TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

KRISTIAN VIRMO

CONTROLLED PRINTING AND COPY PREVENTION IN A QUALITY MANAGEMENT SYSTEM

Master of Science Thesis

Examiner: Prof. Tommi Mikkonen
Examiner and topic approved by the Faculty Council of Faculty of Computing and Electrical Engineering on the 9th of October 2013

TIIVISTELMÄ

TAMPEREEN TEKNILLINEN YLIOPISTO
Tietotekniikan koulutusohjelma
VIRMO, KRISTIAN: Hallittu tulostaminen ja kopioinnin estäminen laadunhallintajärjestelmässä
Diplomityö, 55 sivua, 5 liitesivua
Maaliskuu 2014
Pääaine: Ohjelmistotuotanto
Tarkastaja: professori Tommi Mikkonen
Avainsanat: Laadunhallintajärjestelmä, M-Files QMS, Hallittu tulostaminen, Kopioinnin estäminen, Määrittely

Laadunhallintajärjestelmän (Quality Management System, QMS) toteuttaminen on riippuvainen organisaation prosesseihin liittyvien asiakirjojen luotettavan säilytystavan takaamisesta, säännöllisistä tarkastuksista ja niiden yhteydessä löydettyjen virheiden korjaamisesta. M-Files Corporation kehittää M-Files QMS -laadunhallintajärjestelmää, jonka useat asiakkaat ovat kaivanneet mahdollisuutta seurata, vesileimata ja poistaa vanhoja tulosteversioita käytöstä. Näitä ominaisuuksia yhdistävää toiminnallisuutta, jonka tavoitteena on taata tulosteiden sisällön luotettavuus ja ajantasaisuus, kutsutaan hallituksi tulostamiseksi. Asiakirjojen hallinnalta vaaditaan myös kykyä estää sisällön kopioimista laadunhallintajärjestelmän ulkopuolelle, sillä kaikkien samasta asiakirjasta tuotettujen tulosteiden tulee olla alkuperäisin samasta lähdetiedostosta.

Tämä diplomityö tutkii hallitulle tulostamiselle ja kopioinnin estämiselle asetettuja asiakasvaatimuksia, jotka liittyvät M-Files QMS -tuotteeseen tallennettujen asiakirjojen hallintaan. Työn tarkoituksena on esittää ratkaisuvaihtoehtoja vaatimuksille, valita niistä kelvollisin ja suunnitella sille toteutus. Lopputuloksen kannalta on oleellista, että se ei vahingoita M-Files QMS -tuotteen käytettävyyttä ja hintatasoa. Toteutuksen tulee olla myös jatkokehityskelpoinen.

Ratkaisussa lähestymistapana hyödynnetään M-Files QMS:n räätälöitävän metadatarakenteen ohella M-Files -ohjelmointirajapintaa, käyttöliittymän laajennuskehystä ja ulkopuolisten tahojen tarjoamia ohjelmointikehyksiä. Näiden työkalujen avulla voidaan lisätä vesileimoja tulosteisiin ja rajata myös niiden kopiointia. Räätälöity metadatarakenne mahdollistaa tulosteiden tilan seuraamisen myös levityksen jälkeen. Arvioinnin lopputuloksena asiakirjojen hallitun tulostamisen ja kopioinnin estämisen toteuttaminen M-Files QMS -tuotteeseen havaitaan kannattavaksi, sillä valittu ratkaisu mahdollistaa vaatimusten toteuttamisen edullisesti. Lisättyjä toiminnallisuuksia kuten vesileimausta, voidaan tarvittaessa hyödyntää myös muissa M-Files -alustan tuotteissa. Ominaisuuksien lisääminen on suunniteltu osaksi M-Files QMS:n version 3.0 toteutusprojektia, jonka odotettu julkaisuajankohta on vuoden 2014 viimeisellä neljänneksellä.

ABSTRACT

TAMPERE UNIVERSITY OF TECHNOLOGY
Master's Degree Programme in Information Technology
VIRMO, KRISTIAN: Controlled Printing and Copy Prevention in a Quality Management System
Master of Science Thesis, 55 pages, 5 Appendix pages
March 2014
Major: Software Engineering
Examiner: Professor Tommi Mikkonen
Keywords: Quality Management System, M-Files QMS, Controlled Printing, Copy Prevention, Specification

Verification of process related documentation is an important factor when implementing a Quality Management System (QMS) in any organization. The verification ensures that documents are stored in a reliable location, audited, and corrected in cases where inconsistencies are found. A popular customer request for the M-Files Corporation's M-Files QMS application is the ability to identify and dispose printouts of obsolete versions. This is made possible with metadata based watermarks and tracking objects. These document control characteristics are called controlled printing, and they aim at ensuring that quality documents serve their intended purpose, to verify that all circulated documents are up-to-date and that printouts are generated from the same source files. Copy prevention of documents ensures that only one instance of each document exists in the Quality Management System for printing purposes and that sensitive information does not leave organizational boundaries, copying must also be regulated.

This thesis investigates functional and non-functional customer requirements for controlled printing and copy prevention of quality documents stored in M-Files QMS. The goal is to compare several suitable solutions, select the most fitting option, and design an implementation that is based on it. It is imperative that the solution does not deteriorate the usability and cost-efficiency of the M-Files QMS product, and also that the resulting design is extensible for future upgrades.

The most versatile option was found to be the extension of M-Files QMS with the help of the M-Files API, User Interface Extensibility Framework and references to 3rd party frameworks, while using the built-in metadata structure customization features of M-Files QMS to store the complex variations needed for the implementations. The listed tools are used to apply metadata based watermarks to printed documents and restrict the copying of documents while the built-in metadata handles the storage of tracking objects that enable the tracing of circulated printouts. Based on the evaluation of the selected solution, implementing the required controlled printing and copy prevention features will be worthwhile to the M-Files Corporation as all of the requirements can be met with a minimal extra cost. Also the implemented watermarking features can be reused with other M-Files QMS functionalities. The implementation project has been planned to be included in the development cycle of M-Files QMS version 3.0, which is scheduled to be released in the fourth quarter of 2014.

PREFACE

This master's thesis has been written for the M-Files Corporation as a part of a research project that studied different ways of extending the M-Files Quality Management System product's printing and document tracking features. The purpose of the thesis is to act as a basis for a technical specification related to the implementation project.

I want to thank Professor Tommi Mikkonen who acted as the thesis' examiner especially for helping me find a meaningful structure to the thesis and shape the contents, and also Veikko Juusola for reviewing the thesis on behalf of the M-Files Corporation. I also want to thank Antti Nivala and Mika Salonoja for coming up with the research topic and for the possibility to write the thesis for the M-Files Corporation. Finally, thanks to Merja Virmo for proof-reading the material, and to the Quality Management System development team for the ample supply of design ideas.

In Tampere, 13th of February 2014

_____

Kristian Virmo

CONTENTS

1 INTRODUCTION

To minimize costs that result from production mistakes and customer reclamations, organizations that provide products or services aim to guarantee that the delivery process achieves maximum customer satisfaction while simultaneously minimizing the overall costs of manufacturing and delivery processes. This approach is regarded as Quality Management, and the collection of guidelines which helps organizations in implementing it is called a Quality Management System. The achievement of these guidelines is dependent on accurate, reliable and up-to-date documentation that describes each of the major processes that contribute to the end-product or service. As the contents of these documents are related to the performance of important processes, they have to include a document control scheme that restricts their creation, modification, circulation and disposal. A major requirement is to guarantee that circulated documents are always updated when new versions become available in order to prevent persistence of old information. As an increasing number of organizations have begun to store most of their documents in electronic format, also Quality Management Systems are facing a change, creating new challenges related to these requirements.

The objective of this thesis is to design an implementation that enables two important document control related features on an ISO 9001:2008 compliant, electronic Quality Management System, called M-Files QMS. The first feature is regarded as *controlled printing*, which contributes to making it possible to restrict printing of specific documents to authorized personnel and allowing the generation of printouts that are watermarked and traceable. The second feature is called *copy prevention*, which involves inhibiting users' ability to copy documents regulated content outside of the system, preventing users from printing the contents of documents from unauthorized copies.

Having these abilities in a Quality Management System enables document managers to effectively monitor the circulation of printed versions of regulated documents and recall them when necessary. These features have been demanded by a large group of the current and prospective M-Files QMS customers which makes them high-priority extensions to the current implementation. These customers range from small companies to multi-national organizations, indicating that implementing these changes is important regardless of the customer's size.

The implementation process performed in this thesis includes compiling a list of customer requirements, investigating solutions that achieve these requirements, comparing the solutions and selecting the most suitable option, and finally designing and evaluating the means of adding the features to M-Files QMS. The evaluation phase includes

implementing a small showcase application to substantiate that the solution works in practice. These steps are necessary in order to guarantee that many approaches are investigated and that all of the requirements can be accomplished.

The thesis is divided into 6 chapters. Chapter 2 explains background information related to Quality Management, Quality Management Systems, ISO 9001:2008, Document Control, and software necessary to the solutions. Chapter 3 lists the functional and non-functional controlled printing and copy prevention related requirements. The chapter describes the contents of each requirement and also clarifies why they should be implemented. Chapter 4 introduces solutions that accomplish the compiled requirements. As the study includes two features, these solutions are categorized either under controlled printing or copy prevention. In the end of the chapter, the solutions are summarized and one of them is picked as the basis for the rest of the study. Chapter 5 focuses on evaluating the selected solution. The evaluation consists of use cases that are used to validate that the solution can handle common ways of using the features. The chapter also evaluates the solution against all of the requirements, this time from a technical viewpoint. The chapter ends with the introduction of a sample implementation of some of the requirements. Chapter 6 concludes the thesis by summarizing the evaluation process examining the feasibility of the implementation. The possibility of future extensions is also studied in this chapter.

2 BACKGROUND

This chapter presents the technological and theoretical background information related to the scope of this thesis. Section 2.1 explains the importance of Quality Management, continues with the ISO 9001:2008 Quality Management Systems standard and lastly introduces the related Document Control procedures. Section 2.2 examines the technical aspects and functionality of the M-Files Quality Management System application. Section 2.3 continues by describing the terms Controlled Printing and Copy Prevention. Finally, Section 2.4 introduces one of the candidate platforms for implementation: the WatchDox Document Control service.

2.1 Quality Management

This Section defines the term Quality Management and defines necessary topics related to the subject matter. Subsection 2.1.1 starts by explaining what Quality Management stands for. Subsection 2.1.2 then continues to introduce the Quality Management Standard ISO 9001:2008 and the term Quality Management System. Subsection 2.1.3 concentrates on the documentation requirements of ISO 9001:2008 and Subsection 2.1.4 finally defines the Document Control procedures related to ISO 9001:2008 documentation.

2.1.1 General

According to the glossary of the American Society of Quality, Quality Management is defined as "The application of quality management system in managing a process to achieve maximum customer satisfaction at the lowest overall cost to the organization while continuing to improve the process" [Rumane, 2010, p.24].

Quality Management practiced by businesses does not only aim to make sure that their products and services are of good quality, but also to ensure that an organization's management, products, and services are consistent. Consistency can be achieved by not only focusing on the quality of the end-product, but also planning a method to achieve it and controlling that the implemented plan is followed with reviews. In cases where the plan is found inadequate, necessary improvements are made [Rumane, 2010, p.41].

These steps are taken to prevent costs that come from errors in the, for example, manufacturing processes or services. A business whose manufacturing processes have inadequate levels of Quality Management might for instance repeatedly send wrong bills and inferior products to their customers. Such mishaps become very costly as their number grows and customer satisfaction suffers dramatically. Paying attention to quali-

ty plays a major part in a business' success as organizations can benefit greatly in highly competitive markets if they can maintain a high level of quality while keeping their product and service prices on competitive levels [Aveta Solutions, 2013].

Quality Management becomes a critical issue in businesses when the size of the organization grows and managing personnel and products become more difficult to handle due to the amount of employees and possibly the requirement to move from a centralized leadership model to a more distributed one. For example, a case study carried out in Portugal on a growing mid-sized construction company stated that previously well-organized leadership became less organized and more informal as a result of the company growing in size. The communication between employees and managers suffered as a result of the expansion which led the employees to adopt undocumented informal procedures that slowly began to co-exist with official procedures. The problem with this development was that it was difficult to introduce new processes and procedures as employees assumed that also the informal procedures would be valid. The organization was in a position where it could not guarantee that well documented procedures were being used and as a result could also not guarantee a consistency in their quality of service. To fix the issue, a decision was made to implement a proper Quality Management System [Cachadinha, 2009, p.33].

Quality Management Systems define the way of implementing Quality Management in an organization. They include everything from the way the organization has been structured, what procedures are written, what kind of processes are used and also what resources are needed in implementing a continuous cycle of improvement for both the organization's quality of service and products. The purpose of a Quality Management System is to establish a vision, goals, and a set of standards for employees, build motivation within the company, fight resistance to change and direct corporate culture [International Organization for Standardization, 2012, p.3-5].

Some examples of Quality Management Systems are the process centered ISO 9001:2008 and long term management centered Total Quality Management. The focus in this thesis is on the ISO 9001:2008 family of standards as they are commonly used and have a focus on processes and documentation.

2.1.2 ISO 9001:2008 Quality Management System

ISO 9001 is an internationally recognized Quality Management System standard created by the International Organization for Standardization and ISO 9001:2008 is its latest version. ISO 9001:2008 does not define what the actual quality in services and products is, but instead offers good business practices that help organizations achieve consistent results and continually improving processes [The 9000 Store, 2013A].

ISO 9001:2008 introduces eight management principles that represent the factors that lead to achieving quality. A Quality Management principle is defined by the International Organization for Standardization *as a comprehensive and fundamental rule or belief, for leading and operating an organization, aimed at continually improving performance over the long term by focusing on customers while addressing the*

needs of all other interested parties. Principles that are not followed accordingly lead to less effective operations and inevitably to a failure in quality that can in worst cases cause major harm to individual people or even whole countries [Hoyle, 2009, p.9]. The eight ISO 9001:2008 principles are listed and described in Table 2.1.

The ISO 9001:2008 standard includes the principles *Process Approach* and *Systems Approach to Management* that together characterize organizational tasks as a chain of interlinked processes. A single process is a set of activities that uses resources to transform inputs into outputs. In a System Approach, a process chain is formed as the output of one process becomes the input of another. A Process and Systems based Quality Management System identifies key processes, defines quality standards for them, decides how quality is measured, documents the steps to achieving the desired quality level, and finally evaluating the level of quality and afterwards continuously improving the core processes [The 9000 Store, 2013B].

Table 2.1. *The eight Quality Management principles [Hoyle, 2009, p.10-13].*

Principle	Description
Customer focus	Understanding customer needs and expectations, meeting their requirements and measuring satisfaction.
Leadership	Establishing a clear vision and direction for the organization with a purpose to maintain an environment where the employees are fully involved.
Involvement of People	People on all levels are the essence of the organization. Their full involvement enables all of their abilities to be used to the benefit of the organization.
Process Approach	Desired results are achieved effectively when related resources and activities are managed as a process.
Systems Approach to Management	Identifying, understanding and managing interrelated processes as a system in order to contribute to the organization's effectiveness in achieving its objectives.
Continual Improvement	The continual improvement of the organization's overall performance is a permanent objective.
Factual Approach to Decision Making	Effective organizational decisions are based on analysis of accurate and reliable collected data and information.
Mutually Beneficial Supplier Relationships	Organizations and their suppliers are interdependent. A mutually beneficial relationship enhances the ability to create value for both parties.

2.1.3 Documentation in ISO 9001:2008

Ensuring that the chain of processes in the Systems Approach is defined, maintained, communicated, measured, reviewed and improved is one of the basic requirements for successful Quality Management and relies on up-to-date documentation. In order for a Quality Management System to be ISO 9001:2008 certified it needs to address documentation requirements and also the required procedures associated to them. The different Quality System Document types can be seen in the pyramid shown in Figure 2.1.

The most important Document and the basis for all other documents is the Quality Manual. It explains the scope of the Quality Management System, all of the ISO 9001:2008 requirements, refers to quality procedure documents and includes a flow chart of process interactions. This information is used to communicate the management's expectations for quality to the whole organization. The second level of the pyramid includes documentation of procedures that describe how processes are to be performed. The descriptions state the actor who performs the processes, what is the sequence, and what are the criteria that must be met. The third level includes work instructions that describe how to perform a specific task. The fourth level of the pyramid consists of records and forms that include data and activities that provide evidence of conformance to ISO 9001:2008 requirements [The 9000 Store, 2013C].

To keep these categories of Quality Documents monitored and constantly updated, ISO 9001:2008 requires the Quality Management System to include six procedures: *control of documents*, control of records, internal auditing, controlling nonconforming products, corrective actions, and preventive actions [The 9000 Store, 2013C]. The controlled printing and copy prevention features that are implemented are directly related to the steps dictated in the control of documents procedure.

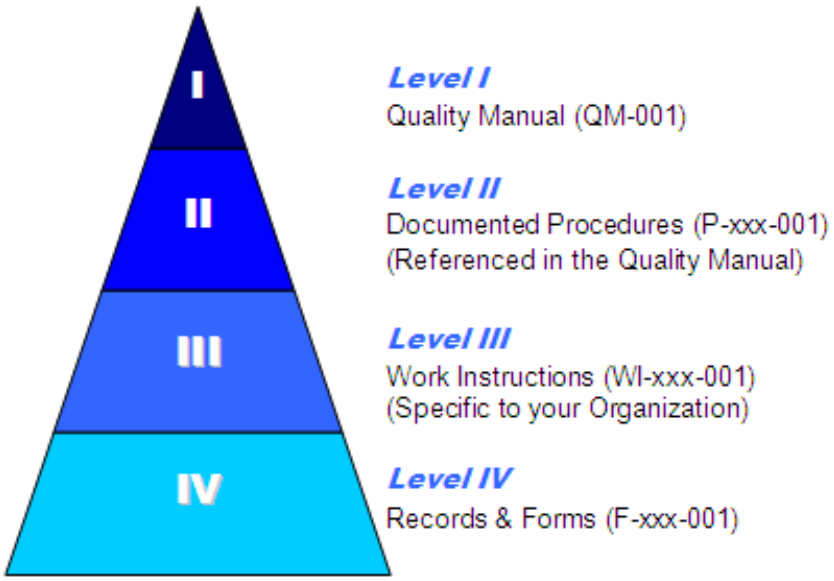

Figure 2.1. *QMS Document pyramid [The 9000 Store, 2013C].*

2.1.4 Control of Documents

Control of Documents means the responsibility of controlling organizations documentation. The ISO 9001:2008 standard states that *documents required by the quality management system have to be controlled* [Hoyle, 2009, p. 250].

A major part of ISO 9001:2008 certifying a Quality Management System is to introduce a strict document control process. Quality Managers need to decide what kinds of documents are taken into the process, how they are maintained and how possible nonconformities are handled in the Quality Management processes [Cachadinha, 2009, p.35-56].

A document is data or information that has been stored in either paper or electronic format. Documents required in an organization's Quality Management processes are the ones that are created in any of the steps that are included in the current Quality Management System. The ISO 9001:2008 standard clause 4.2 specifies that a fundamental principle of quality assurance is that the organization in question is prepared to substantiate by objective evidence that they have maintained control over activities affecting the quality of products supplied to customers. This objective evidence is formed by three components:

1. Documented policies, plans, processes, standards, and procedures that declare intensions and planned arrangements for meeting customer requirements.
2. Records proving that intentions and arrangements have been implemented.
3. Records that the supplied products or services meet customer requirements.

These documents always require special attention that is called Document Control [Hoyle, 2009, p.255].

Document Control is the process of regulating an organization's documents, and defining the steps of development, approval, issuing, changing, distribution, maintenance, use, storage, security, obsoleteness, and finally disposal by some mean. Through these steps control of documents aims at ensuring that the documents serve a purpose in the quality system, that non-essential material is not distributed, that employees have access to necessary information related to their work, that access to classified information is restricted, and that distributed material stays up-to-date [Hoyle, 2009, p.250]. An example of one Document Control life-cycle is visualized in Figure 2.2.

The ISO 9001:2008 standard states that *Document Control Procedures* are written to define the controls that are applied to required documents that are essential in achieving the desired level of quality. This means that there must be recorded guidelines to performing each Document Control activity to ensure that there is consistency and predictability in generating documents. Some examples of information that the Document Control Procedure should require of all controlled documents are: layout format, list of people responsible for the preparation steps, *limitations in use and distribution, list of employees with printing authorization, steps for following document circulation*

and disposal. This information enables easy comparison between documents generated from the same process and allows the improvement of steps that are deemed as inadequate or ineffective [Hoyle, 2009, p.251-252].

Before Electronic Document Management Systems were taken in to use, documents have been handled in a manner that paper documents have been stored in a physical archive. The documents were then copied using a photocopier of the original document. These copies were distinguished from the original ones with various stamps and watermarks and then distributed to the target group. Now that many organizations have been converting their old Quality Management Systems from paper-only record implementations to electronic versions, also document control processes have to be implemented in the new Electronic Quality Management System [Hoyle, 2009, p.252]. One example of an Electronic Quality Management Systems is the M-Files QMS.

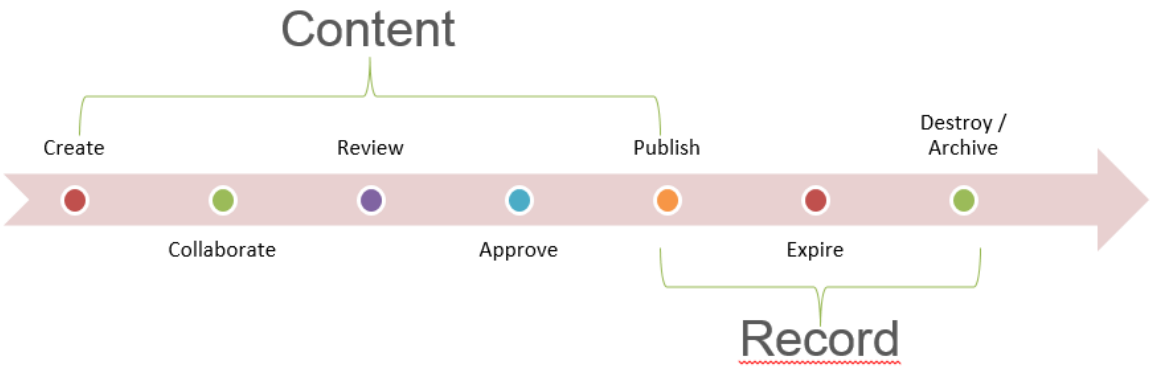

Figure 2.2. *Quality Document life-cycle [M-Files, 2013D].*

2.2 M-Files QMS

This Section introduces the M-Files QMS application and its main features. Subsection 2.2.1 starts with general information about the product and Subsection 2.2.2 explains its technical aspects. Subsection 2.2.3 lists the different metadata types and their uses. Subsection 2.2.4 shows what functionalities the M-Files QMS client application's user interface offers and Subsection 2.2.5 introduces how document control has been implemented. Finally Subsection 2.2.6 explains what tools are available to the developers of M-Files QMS.

2.2.1 General

M-Files QMS is a Quality Management System developed by M-Files Corporation as an extension to the M-Files Electronic Document Management System [M-Files, 2013A]. The latest version in development at the moment of writing is 2.0 and the description is written based on its list of features. M-Files QMS is available only for the Microsoft Windows operating systems, but its basic features can be accessed also from other operating systems via a web-interface or a mobile client. The software is used to control quality documentation, personnel training and to follow compliances and regula-

tions. M-Files QMS has functions that help automate tasks that are repetitive, with organizing and managing Control of Documents and also collection and use of electronic signatures [M-Files, 2013B].

M-Files QMS introduces features that assist organizations in following Quality Management Standards such as ISO 9001:2008. The features that aid in following the standard are: marking or labeling stored documents, producing electronic signatures and auditing using these features together [M-Files, 2013A].

The defining features of M-Files QMS compared to other Quality Management Systems are the importance of stored metadata and its basis on the M-Files Document Management System which has broad document storage and versioning capabilities. All files and other objects stored in the QMS are paired with a collection of meaningful metadata that describes the content state. This metadata can be taken advantage of, for instance when searching for or sorting documents and other stored objects [M-Files, 2013B].

2.2.2 Technical features

M-Files QMS is based on the Client-Server model in which users have installed a client application that connects to a main server that houses the server application and handles the storage of files and metadata. The Client-Server architecture and the compatible operating systems and devices are demonstrated in Figure 2.3. There are various client application options such as a native Windows M-Files client, a web browser client and also mobile device clients for different smartphone and tablet operating systems. M-Files client works natively on the Windows platform, but other operating systems such as Mac OS and Linux can access the server content from the M-Files Web Access client. Web Access acts as a similar access point to the M-Files QMS server as the Windows client, but via a web browser. Currently the supported list of web browsers consists of the latest generations of: Internet Explorer, Mozilla Firefox, Google Chrome, Safari and Opera [M-Files, 2013B].

The server application can be installed on a Windows server, cloud service or a Windows desktop. Administrative users can connect to the server software by using a Server Administrator application that offers a user interface for performing various administrative tasks [M-Files, 2013B].

Physical storage of file contents and metadata is handled on the server-side of M-Files. The storage resides either on a Firebird or Microsoft SQL Server database that can be set up on the same server or an external device. The client application has a temporary cache that stores frequently accessed data and makes repeated tasks faster to accomplish. The Windows M-Files client application accesses the database contents from the server and displays them as a mounted hard drive in the Windows Explorer allowing data to be fetched by using Windows file system commands [M-Files, 2013B].

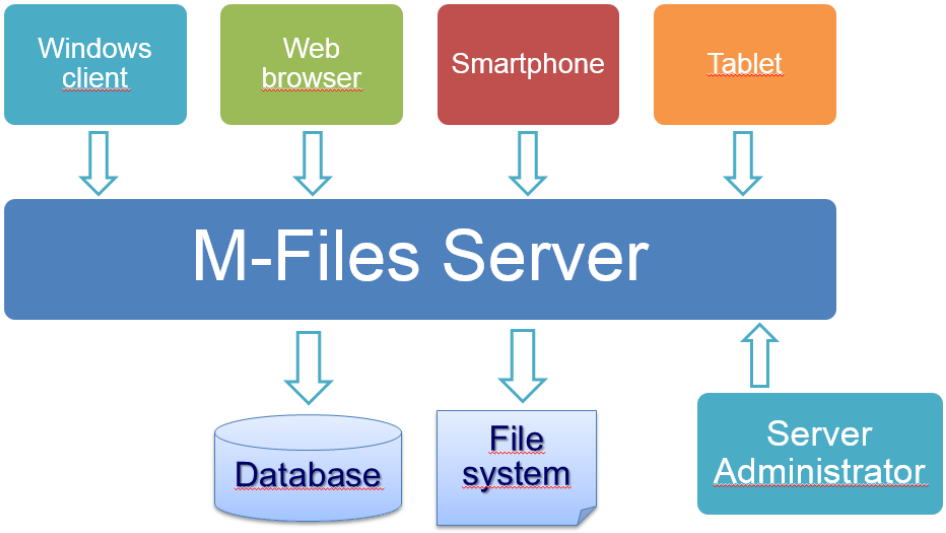

Figure 2.3. *M-Files Client-Server Architecture [M-Files, 2012A].*

2.2.3 Metadata types

The metadata in M-Files QMS is information that describes the contents and use of a document or other non-document object. This information can be for example date of creation, specifics of a contract, or the recipients of a letter. Metadata can be used as a basis to sorting documents or objects and as search parameters.

The information in M-Files QMS can be accessed from a storage instance called a *vault*. There can be multiple different vaults for different purposes and all of them can be set up with different object types, classes and properties. There is also a possibility to replicate content or metadata between vault instances. The number of M-Files QMS vaults increases usually based on the size of the organization. Small companies might have needs for only one document vault that stores relatively few documents, whereas a large organization needs to store millions of objects. In such a case it is also important to split the vaults for example based on departments (Research and Development, Marketing and Consulting).

A user created item that is stored inside M-Files QMS is called an *object*. Objects are individual instances of o*bject types* that act as rough stereotypes that are used to sort information. *Document* is the default object type that is meant for storage of files. It is one of the object types that appear in every vault created in M-Files QMS. Other object types such as Projects, Customers or Contact Persons can also be created on demand. The M-Files server administrator can decide if any of these object types can contain files.

Information and files contained in object types are sorted by *classes*. Classes are assigned to every object and act to further categorize the objects based on their use. Classes include metadata fields that can either be required or optional. For example a document could have classes for a public memo and an internal memo where the public memo has a requirement for customer information.

The metadata fields that are attached to classes are called *properties*. Properties have varying data types such as free-form text, numbers, object references and dates. Administrative users can create and modify object types, classes and properties. The Server Administrator user interface and the various customizable metadata types can be seen in Figure 2.4.

Access to documents and other objects are controlled by user *permissions*. Users can add or modify permission rules to objects that are either created by them or ones that they currently have access to. Permission rules include *read, edit, change permissions and delete* categories that can be set either to individual users or user groups. Complex permission settings can be grouped under *Named Access Control lists* that can be reused and prevent redundancy.

Objects stored in M-Files QMS can also include a *workflow* and a *workflow state*. These represent processes and the transition between the process phases. Workflows can be created from the administrator application where they can be customized to perform specialized actions such as sending notification e-mails when a state transition has occurred. The states are then assigned to objects and controlled either from the client or from the M-Files QMS API. These notifications and interactive elements make it possible to create complex automated tasks within the M-Files QMS vaults.

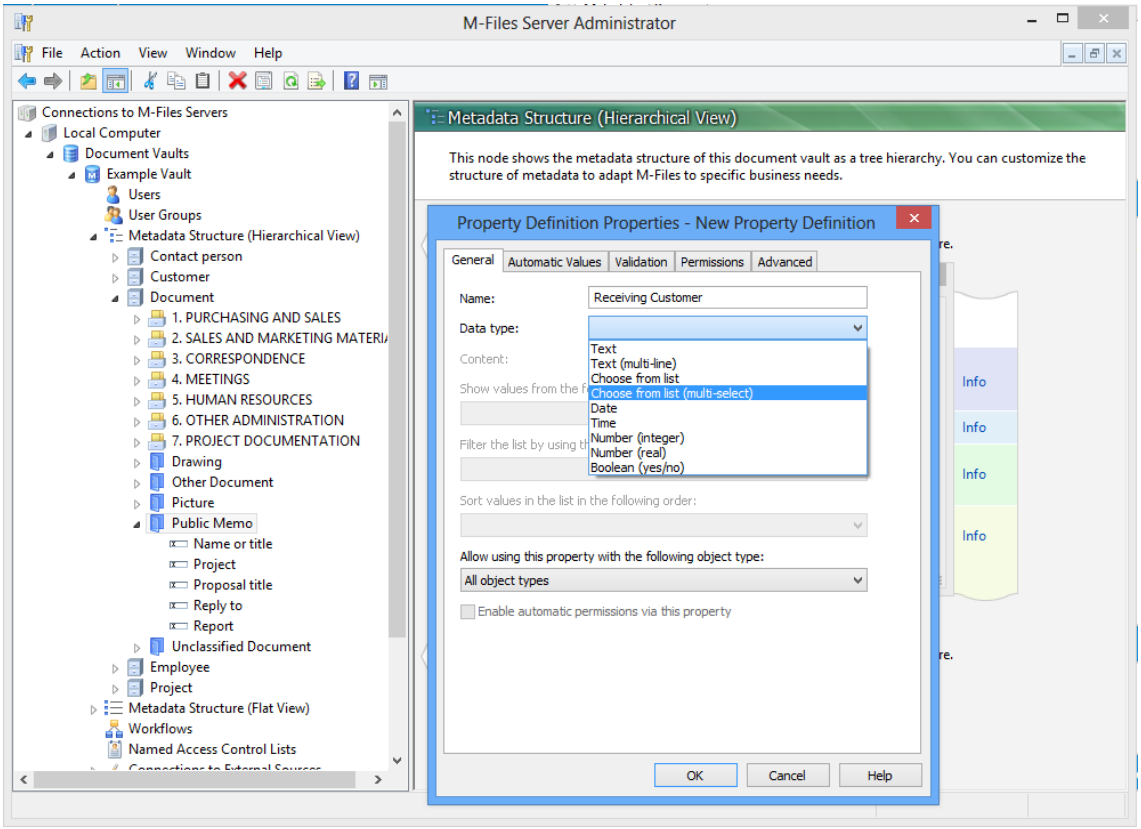

Figure 2.4. *M-Files Server Administrator application.*

2.2.4 M-Files QMS Client User Interface

The most important thing about the user interface of M-Files QMS client application is its aim to be easily approachable by all kinds of users. The client application integrates directly to the Windows Explorer interface and behaves similarly to the standard Windows folder structure, so the learning curve for existing Windows users is minimal. Even though the interface aims to perform similarly to the Windows interface, the handling of file contents and metadata of documents and other objects is handled differently. As M-Files QMS works as a version control tool, the user must first check out the content to be able to modify it. When an object is checked out, it cannot be modified by other users until it is checked back to the vault with the changes. Once the object has been checked in, a new version is created to represent the new modifications. This feature exists to protect users from losing content due to concurrent modifications as the same version cannot be altered simultaneously by multiple users [M-Files, 2013B].

The M-Files QMS client application user interface splits into three main sections: the task pane on the left side, content listing in the center area and the metadata card area on the right side. These different sections of the interface can be seen in Figure 2.5. The task pane area contains shortcuts to commonly performed actions such as creating new documents and other objects, manipulating the content by, and also links to operations that can be performed on currently selected objects. The shortcuts and links in the task pane area can be customized for all vault users simultaneously by an administrator, or for personal use by an M-Files QMS user. The content listing displays the contents of the currently opened view or search result which are divided by object types. The listing area enables users to browse through found content, view contents and relationships of objects, and also select objects for modification. The metadata card displays the class, properties, permissions, and workflow states of the currently selected content listing item. The main purpose of the card is to enable users to either add or alter object metadata when a new object is created to the vault or an existing object is being modified.

Objects can be found from the M-Files QMS vault by utilizing one of two methods: searches and views. Searching the current vault for documents and other objects is done by using the search bar on the top of the window. All searches are based on metadata values, such as object name, class, object type, or the user that created the object. It is also possible to use search parameters based on file contents. Objects in the vault can be further filtered by creating views that act as substitutes to the traditional Windows folders. These filters are similar to the parameters that are used with the object searches. An M-Files QMS user could for example create a shared or private view, that contains document objects and sorts them into sub-views called virtual folders based on the related customers.

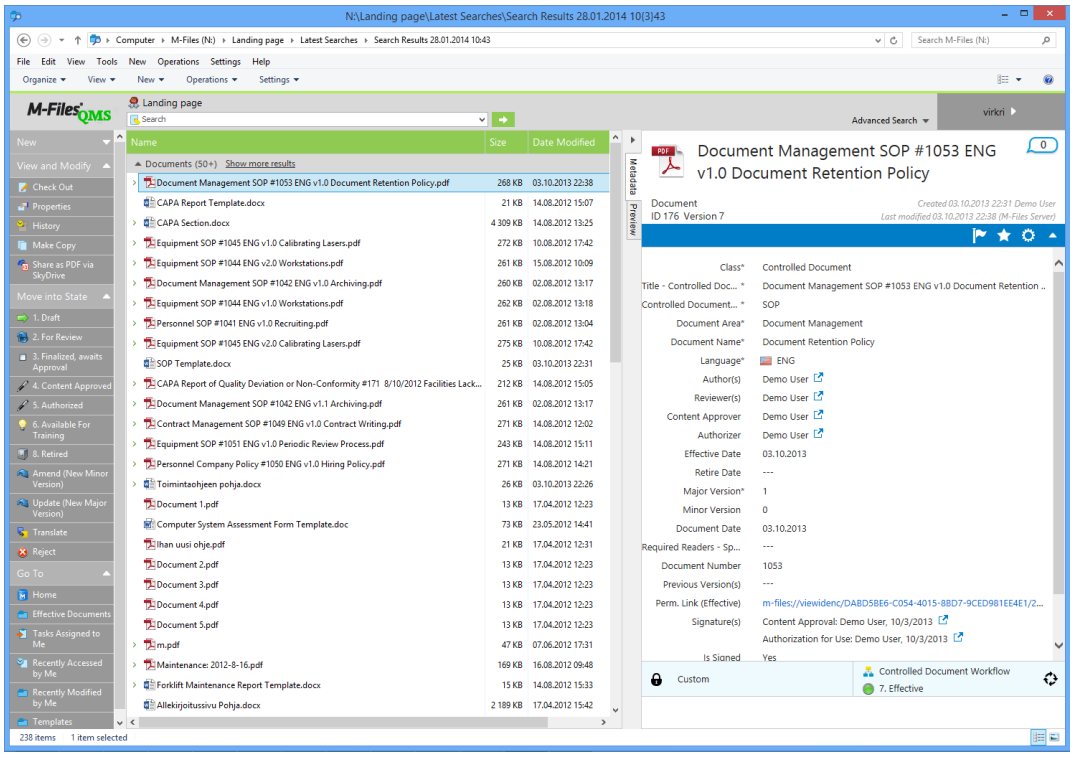

Figure 2.5. *M-Files QMS Windows client application.*

2.2.5 Controlled Document Workflow

Documents that are stored in M-Files QMS and required by the Quality Management System are represented by a custom object type with the name: *Controlled document*. Documents stored under this object type are essential quality-related documents such as standard operating procedures, manufacturing instructions, safety documents or company guidelines. In an unmodified M-Files QMS environment all of the controlled document objects use the specifically created controlled document class. As a major change compared to other default and custom classes in M-Files QMS, the controlled document class has special features that help document managers in controlling the document control workflow, visibility and publishing of essential quality documents. Significant differences to normal object classes include automatic document naming and numbering policies, electronic digital signing automation, automatic PDF conversion and also dynamic permissions that are related to the current state of the document control workflow [M-Files, 2012D, p. 42].

All controlled documents are identified by three different version categories: major versions, minor versions and M-Files internal versions. The major version number identifies the originally published version of the document. A new major version number means that the controlled document has undergone major changes and has been effectively republished. Minor versions indicate small improvements made to the controlled document's major version. These improvements can be, for example clarifications and footnotes to the original document. Internal versions are the built-in M-Files identifiers that are unique to each stored object in the vault [M-Files, 2012D, p. 42-44].

The steps that occur in the lifetime of a controlled document start from the creation of the original draft version, and continue to being authorized or rejected. After authorization, the controlled documents are published and after a certain effective time they finally retire. The progress of the workflow from the drafting phase to retirement is visualized in the diagram shown in Figure 2.6. The path from state 1 (S1) to state 8 (S8) is contained under one major version and each state change between them creates a new minor version of the document. After the initial drafting phase, the document is audited and either approved or rejected based on the results (S3, S4 and S5). The approved document is finally authorized and set available for training purposes. All of the employees under the training requirement have to learn the document (S6) and sign it digitally for it to be moved to effective use (S7). A new major version can be created in the form of rejecting, amendment and updates (S9, S10 and S11). After the document is deemed obsolete by the document manager, it is moved to the retirement state (S8). This includes making the document inaccessible from M-Files QMS and also the collection of possible paper printouts [M-Files, 2012D, p. 57].

.

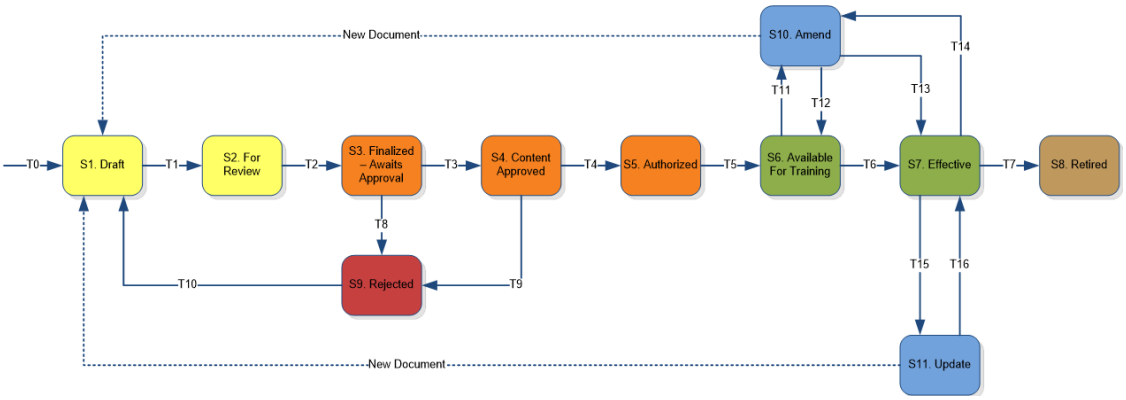

Figure 2.6. *M-Files QMS Controlled Document workflow diagram [M-Files, 2012C].*

2.2.6 Development tools

M-Files QMS offers extensive possibilities for customization as it includes an integrated application programming interface (API) and also the M-Files User Interface Extensibility Framework. These development tools allow software designers to both customize the user interface and create server-side modules that expand on the standard functionality of M-Files QMS.

The M-Files API is a collection of classes and operation interfaces that offer access to M-Files QMS from various scripting tools and environments. Possible uses include fetching of file contents and metadata from M-Files QMS vaults or integrating a 3rd party application with M-Files QMS. Developers can access and manage objects that are contained in an M-Files QMS vault. This means that operations such as creating, viewing, modifying, destroying and also controlling the visibility of objects is possible. The API is an ActiveX/COM DLL and it can be used with Visual Basic, VBScript, C++, and all other .NET languages such as C# [M-Files, 2013C].

The M-Files User Interface Extensibility Framework is a collection of features, APIs, program execution environments, and libraries that allow the developers to customize the appearance and functionality of the M-Files QMS client application. The modifications are written as applications that are executed alongside the client application in order to change the M-Files experience to match a specific business area or need. Example use cases of the User Interface Extensibility Framework include the modification of the interface by altering the appearance of the task pane, content listing or the metadata card. In addition to the change in appearance, also the core functionality can be customized to match the needs of the developer. Physically the application modules are implemented as one or more Jscript files that have access to interfaces that allow communication with the M-Files QMS platform [M-Files, 2012B].

2.3 Document control requirements

This Section introduces necessary document control functionalities that are required in the M-Files QMS application. Subsection 2.3.1 explains what controlled printing stands for and what are its requirements while Subsection 2.3.2 introduces the term copy prevention and its significance in document control.

2.3.1 Controlled Printing

ISO 9001:2008 is the Quality Management standard that is most common among M-Files QMS customers which means that documenting processes and procedures is a requirement for M-Files QMS and with it comes a need for continual review and improvement. ISO 9001:2008 dictates that all essential quality documents and records have to be regulated, ensuring that printouts created from these are labeled, traceable and also retrievable for disposal in cases where the information has become obsolete or a new version is ready for distribution. In highly regulated environments such as the healthcare industry, controlled documents require special attention as they might contain sensitive information such as customers' personal information. Documents handled by high technology organizations also include data that falls under intellectual property rights. Both of these examples rely on the fact that the state and location of printouts created from secure electronic content are always known.

As constant access to the used Quality Management System cannot always be guaranteed, paper printouts are still a favored approach by various organizations for distributing information inside and outside the organizational boundaries. According to M-Files Corporation sales and consultant divisions, over half of current and potential M-Files QMS customers are requesting a possibility of controlling who is able to print controlled documents and afterwards keep track of all of the printouts. This request for more control in printing is directly related to the document development and maintenance rules of ISO 9001:2008. These rules state that as most organizational processes are likely to include and require some form of controlled documentation, the used procedures for process documentation should also include information on who is able to

print, publish, use, distribute and dispose of said documents [Hoyle, 2009, p.252]. Such attention is vital in ensuring that sensitive material does not leave organizational boundaries and also that old versions of distributed printouts are being tracked and retrievable whenever they need to be recalled.

Having the ability to control printing and tracking the printouts built in the application is an important feature when implementing an effective and reliable electronic solution as the resulting automation replaces many steps that required routine work done by document managers. An example of a category of documents for which this addition is especially important is the Standard Operating Procedure (SOP) document. SOP documents include guidelines that explain how machines operated by employees are to be used safely and effectively. It is essential that all employees have read and understood the contents of the latest effective version of the SOP related to their workstation before they can either begin or continue operating it. To ensure that used SOPs match the workstation and its location and that they are of the latest version, the printouts have to include a unique identification number, version number, effective beginning and expiring dates and language codes. The SOP documents become obsolete once major changes are made to the content, and in this event they are recalled and finally destroyed by the document manager. Tracking based on the identifiers created by automation make this an efficient process [M-Files, 2013E].

2.3.2 Copy prevention

Controlling access to data and information that is mainly contained in documents in order to protect sensitive information from unauthorized access is called Information Rights Management. An example of Information Rights Management can be found in the Microsoft Office product family that offers security and copy protection settings to document types such as Word documents, workbooks and presentations. Another similar application is built-in to PDF documents [Microsoft, 2013].

Information Rights Management in software such as Microsoft's Office product family is implemented to prevent unauthorized users from forwarding, copying or modifying content. Use of documents, workbooks and other content can also be regulated by setting a specific timeframe during which they can be accessed. These options allow organizations to set and enforce use and dissemination policies, for example when dealing with sensitive controlled quality documents. These restrictions only apply to actions that users might target towards the protected documents, malware such as Trojan horses do not fall under the scope of the security settings and must be taken care of with 3rd party security applications such as virus scanners [Microsoft, 2013].

Security features that enable copy prevention can also be embedded to the file-types themselves. An example of such a scenario is the Portable Document Format (PDF) which includes a variety of security options that allow content creators to specify security measures based on the level of need. Security measures include password protection, encryption, read-only flags, print prevention flags, and built-in copy prevention [Adobe Systems Incorporated, 2013A]. The built-in security features of PDF files can

be accessed with most PDF reader applications, one example being the official Adobe Acrobat X. The user interface of the Acrobat X security settings can be seen in Figure 2.7.

The use of built-in security settings is considered as an adequate level of protection in environments where the organization has a strict control over the applications that the users can install on their work computers. Such supervision is important as there are applications and other methods that can circumvent these settings. Another interpretation for copy prevention is the disabling of file transfer features that allow users from moving the protected file itself outside of the regulated zone. This includes User Interface level actions such as copy-paste, drag-and-drop and also command console operations that are used to move files. Preventing the transfer of the file itself is an additional counter-measure against data-theft, as the security settings of Information Rights Management can be circumvented by using software that does not support these features [Autodesk, 2013].

Figure 2.7. *Security options for PDF Acrobat X [Adobe Systems Incorporated, 2013B].*

2.4 WatchDox

WatchDox is a service that enables organizations to access, share, and control documents on any tablet, smartphone or PC device. It is available both as a Software-as-a-Service System and on-premise installation, and it enables a centralized way of storing, controlling, and even remotely destroying business-critical documents. WatchDox includes an API framework that allows it to integrate with 3rd party applications and allow for the security and document control features to be used outside of the default applications. [Watchdox, 2013A].

Depending on the contract, WatchDox acts as a cloud or on-premise service that acts as a storage location for controlled documents. The service gives administrators the possibility to create user accounts for people inside and outside the organization, and then setting permissions for stored documents individually. The permissions include options such as viewing, editing, printing, forwarding, and watermarking. The administrator can then share a link that allows the users to download the document to their WatchDox application and view the contents. As the documents can be viewed only from the client application, the administrator can at any point restrict access to the document, even after the file contents have already been downloaded [Watchdox, 2013B]. WatchDox also includes a number of tools that help with advanced tracking of document activities, as an example document access can be logged based on IP address and geographic location information [TechCrunch, 2009]. A demonstration of the WatchDox user interface listing shared documents and related functions can be seen in Figure 2.8.

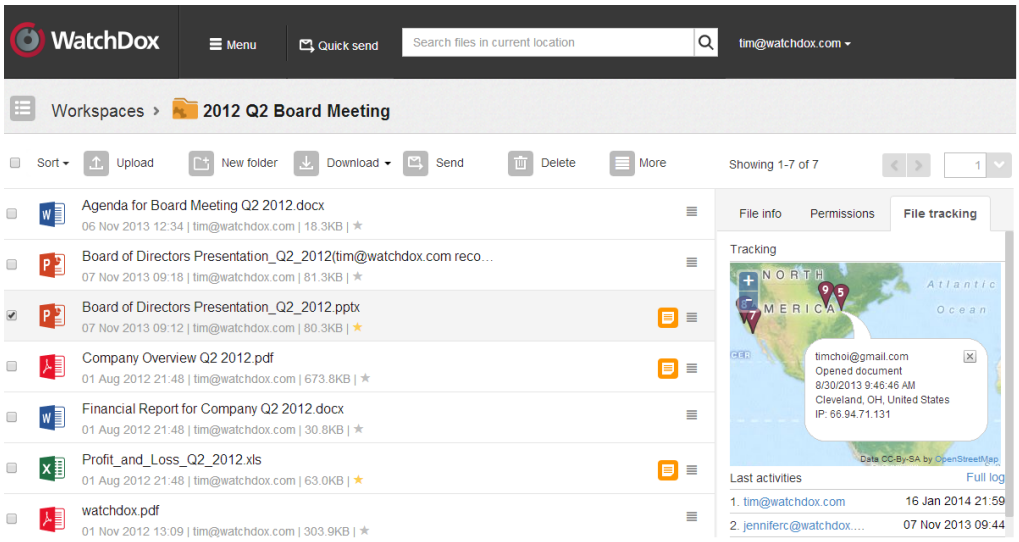

Figure 2.8. *WatchDox document listing [WatchDox, 2013A].*

3 REQUIREMENTS

This chapter describes the desired functional and non-functional requirements for the controlled printing and copy prevention implementation for M-Files QMS. The contents of requirements are detailed, and reasons are given to support their need of implementation. All of the requirements listed in the following sections are targeted only to function with the Windows client of M-Files QMS and not for example the mobile application or web browser client that can also be used to view M-Files QMS vault contents.

Functional requirements are used to define the application's behavior in a specific situation and reaction to user input. Their purpose is to list what services and functionalities the software offers. Functional requirements are covered in Section 3.1. Non-functional requirements set restrictions and constraints to the functional requirements and describe what conditions must be met so that the functional requirements can be implemented successfully [Verkamo, 2004]. Non-functional requirements are covered in Section 3.2. These requirements are collected and refined from the requirement document compiled by the M-Files Corporation consultants and M-Files QMS developers [Appendix 1] and also from a set of sample requirements specified by a prospect customer organization [Appendix 2].

3.1 Functional requirements

M-Files QMS is currently missing an essential document control feature that enables organizations to manage which users have the permission to create printouts of controlled documents and also copy them outside of the M-Files vaults. This feature is split into two separate components called *controlled printing* and *copy prevention* which are to be planned and implemented as new additions to the M-Files QMS 2.0 release. The controlled printing component covers the identification of users who are attempting to create new printouts, logging the transaction and also labeling the created printouts with identifying watermarks and copy prevention covers disabling the downloading of controlled documents outside of M-Files for unauthorized use. Controlled printing requirements are listed under Subsection 3.1.1 and copy prevention's under Subsection 3.1.2.

3.1.1 Controlled Printing

This subsection introduces seven functional requirements for the controlled printing aspect of the implementation project. The requirements are listed in the order of importance starting from the most critical one.

1. Possibility to create printouts from M-Files QMS. The most important requirement for the controlled printing feature is the ability to select a controlled document object in M-Files QMS and then produce a printout out of the latest version of that document. The feature must be functional only with the controlled document objects and specifically ones that have been marked by a managing user as currently printable. There must also be an option for the user to select a printer device and printing related parameters such as paper orientation and color options.

2. Possibility to exclude users from printing a document in M-Files QMS. The possibility to prevent printing of controlled documents based on user permissions and other metadata is also a very important requirement in the implementation of controlled printing. Unless this functionality is implemented, none of the security related requirements are relevant since any user that is logged in to the M-Files QMS instance can generate printouts of controlled documents. This requirement involves identifying if the current user is in a user group authorized to produce printouts and that the user has permissions to print the specific document. If these conditions are not met, a warning needs to be shown to the user, a log entry needs to be written, and the printing operation must be stopped.

3. Possibility to view and edit current restriction rules. Users that manage controlled documents have to be able to alter controlled printing permissions from the M-Files QMS client user interface. The client needs to have an ability to show a list of users who have controlled printing feature authorization and also controls that allow the manager to add or remove users from the list. This requirement ensures that restriction rule modifications can be done from the client application without requiring administrator tools or access to high level functions.

4. It must be possible to track every controlled printout. One of the required features in the standard and the implementation is the traceability of generated printouts. Every controlled printout produced from M-Files QMS must leave a log entry. This log entry can be either a system log level entry that is stored in the M-Files QMS server application's event log or a dedicated M-Files QMS object that is stored in the vault. A combination of both is the best alternative if such a solution is possible. The log entries must contain information of who created the printout, a timestamp of the transaction, and a connection to the exact controlled document that was printed. To guarantee that the printouts can be identified, the controlled document identifier and version numbers need to be attached to the log entries. All printout instances have to have unique identifier numbers to separate them from each other.

5. Producing printouts with custom labeling. The printouts that are produced by the user in M-Files QMS must include some or all of the three labeling types. The first type is static text, such as "Controlled copy", as the document's header, footer,

sidebar or watermark. The second type is metadata based labels, such as the date of printing or the name of the user that produced the printout. The third option includes labels with varying stamps and images such as a logo of an organization. The labels that are included in the printouts can be disabled or enabled in the M-Files client application by a user that has the required permissions. Labeling rules can be set separately for all controlled documents so that they can vary based on the document's content and state.

6. Tracking controlled printouts with M-Files QMS notifications. The controlled printing implementation needs to be able to send e-mails to document managers and other responsible users when controlled printout objects are altered. These situations include the creation of new printout objects, workflow state changes on existing printout objects, and the modification of printout related metadata. The automatic messages must be implemented with M-Files QMS notifications in order to ensure seamless integration with the application.

7. Setting controlled printing as default printing behavior. Currently the default behavior for printing when using M-Files QMS is the Windows printing function. This default behavior must be replaced with the controlled printing function when attempting to print controlled documents in M-Files QMS. The change is required as any attempt to print the controlled documents with the default Windows print action will result in the document contents being printed without watermarks and other labels set by the managing user.

3.1.2 Copy Prevention

This subsection introduces three functional requirements set for the copy prevention component of the implementation project. The requirements are listed in the order of importance starting from the most important one.

8. Preventing copying of controlled documents. When controlled document versions enter a state where they are authorized for use, further editing is no longer possible inside the M-Files QMS vault. Copy prevention for authorized controlled documents needs to be implemented in order to guarantee that the documents stored in an M-Files QMS vault are not copied to external locations, modified or printed without authorization and that the necessary watermarks are not removed. Prevention includes cases where the built-in copy-paste and drag and drop features of Windows are used to copy and move controlled documents from an M-Files QMS vault to the hard drive. The goal is to ensure that all of the paper printouts of documents in circulation have been created from the same source and that they are of the latest version and that old versions can be reliably tracked from the controlled document metadata. This prevention rule also applies to users that have permission to manage controlled documents and their rules in the M-Files QMS vault. This is to ensure that the controlled printing procedure is never broken, even by accident.

9. Creating PDF documents that are not printable with standard viewers. In special circumstances, managing users might need a possibility to hand out a PDF copy of a controlled document to another user. In this scenario, it is necessary that these copies cannot be used to create unauthorized printouts. It is expected that in organizations that use controlled printing, also the use of software is highly regulated and non-sanctioned 3rd party software is not allowed. Therefore PDF documents should be modified to be in read-only mode and non-printable to standard viewers such as Adobe Reader. This requirement does not apply to the built-in Windows copy and paste tools as it needs to be implemented as a new M-Files QMS function that creates a modified copy of the source document.

10. Preventing taking of screenshots of controlled documents. One weakness that occurs from viewing the controlled documents in any kind of PDF viewer is that the operating system makes it possible to take screenshots of the displayed content. This can easily be done in Windows environments by opening the document in a PDF viewer application and then pressing the print screen button on the keyboard. The screenshot can then be opened in an image editing application and saved in picture format. Once the copy is on the hard drive, it can then be printed without authorization and necessary watermarks. To prevent this, a function must be created to hide the shown material from these screenshots. The taken screenshot would in this case show only a black box instead of the part where the controlled document would normally be shown.

3.2 Non-functional requirements

The three non-functional requirements listed in this section apply to both the controlled printing and copy prevention. They are listed in the order of importance starting from the most important one and ending with the least important.

11. Price of implementation. The controlled printing and copy prevention features are to be implemented into an M-Files QMS instance that could have thousands of individual users. As a result, an important part of the solution is that it has to be affordable either on a per user basis or as a fixed cost. The tools and services required in implementing the controlled printing and copying prevention features have to offer benefits that match the price point of both license fees and fixed costs.

12. Usability. The controlled printout creation process needs to be a task that requires no special knowledge or additional training outside of basic M-Files user skills. All users that know how to use the M-Files QMS client application should be able to create printouts with ease. In order to achieve this, the look and feel of the implementation needs to be similar to the current M-Files user interface and should not make the use of the existing features more difficult. Controls that can be interacted with need to

be usable with mnemonic key combinations and also selectable with Windows key commands such as the arrow buttons. The dialogs, text and controls that are used in the implementation have to be localized to match the language version of the currently installed version of M-Files QMS.

13. Extensibility. The controlled printing and copy prevention features must be implemented in a way that allows new developers to easily start working with the existing framework, for example when additional features are to be developed. All code segments have to be written to match the requirements set in the M-Files Corporation internal programming style manual. Public interfaces, classes and methods have to be written in a generic manner which allows them to be used again in later extensions and possible integrations to other M-Files products.

4 POSSIBLE SOLUTIONS

This chapter introduces possible solutions and technologies for implementing controlled printing and copy prevention to support M-Files QMS' document control procedures. Solutions in Sections 4.1 and 4.2 explain what kind of features the solution includes, what its strong and weak points are and why it should or should not be taken into use. The weaknesses and strengths of all solutions are weighed against the functional requirements introduced previously. Finally in Section 4.3 the different solutions are weighed by their benefits and weaknesses and compared against each other in order to find the best possible basis for implementation. Section 4.3 also contains the comparison of the candidates based on their compliance with the non-functional requirements.

4.1 Controlled Printing

This section introduces implementation possibilities for the requirements of controlled printing for quality documents. The method of using the built-in features of M-Files QMS is explained in Subsection 4.1.1. Using M-Files QMS Web Access to as a framework for controlled printing is introduced in Subsection 4.1.2. Using the M-Files QMS API and User Interface Extensibility framework to modify the product to suit controlled printing needs is explained in Subsection 4.1.3. Storing and sharing controlled documents for controlled printing purposes in the 3rd party WatchDox service is studied in Subsection 4.1.4.

4.1.1 Using only the M-Files QMS built-in features

One possibility of implementing controlled printing to M-Files QMS is to use the core product as-is and build a framework that uses only the existing features and administrative users' management to accomplish the requirements.

Implementing functional requirements for printing controlled documents and authorizing users (Req. 1, 2 and 3) using only the current features of M-Files QMS can be done by creating a new object type for controlled documents that can be printed by users. An administrator or a quality manager user has to manually set permissions on these objects to control who is able to see, edit, and produce printouts. Restriction can be implemented by denying users and user groups from reading the printable controlled documents. Current restriction settings can be viewed and edited from the M-Files QMS native permissions dialog. As M-Files QMS supports Windows shell commands, document files that are stored under these printable controlled document objects can be accessed from the file system level, and the Windows native printing dialog can be used to print the contents.

Implementing the possibility to track printouts (Req. 4) to function independently is difficult as printing the contents of a controlled document does not leave an electronic footprint other than a ‘file downloaded’ event log entry. Users would need to contact the person in charge of the controlled document who would then personally keep track of the generated printouts. The main weakness in this method of record keeping is that the person in charge has to monitor every printout and guarantee that unauthorized printouts are not generated without his knowledge, as no automatic electronic entries are stored of transactions. As one of the document control requirements is collection and disposal of obsolete documents, the whole process is reliant on the responsible person’s record keeping abilities.

Currently there is no way to implement printout watermarking (Req. 5) with the built-in features of M-Files QMS. Adding metadata based labeling to the printed controlled documents could be possible with server-side scripting that uses the M-Files API and is triggered by workflow state changes, but there are currently no current API methods that allow manipulation of document file contents in such a way.

Tracking of printouts with M-Files QMS automation (Req. 6) can be accomplished by sending notification messages to the document manager when a specific workflow state is entered. Entering this state also writes an “Object state changed” entry to the M-Files QMS Server event log.

Changing default printing behavior (Req. 7) cannot be done with the built-in features as the Windows printing commands are always available by design in the M-Files QMS client application. Changing this functionality would require major changes in the M-Files QMS core framework and cannot be easily implemented.

The strength of using the built-in features is that for most requirements there is no need to add more functions to the M-Files QMS client application as it already offers a lot of customization options. Using only the existing features would save a lot of time as no advanced scripting would be required. The weakness of this approach is that the implementation will depend on a lot of human interaction and record keeping, which would be more reliable as automatic operations. M-Files QMS was not designed to include user specific restrictions on functions such as printing, so this implementation can also prove to be unreliable.

4.1.2 Modifying M-Files QMS Web Access

M-Files QMS has a Web Access user interface that can also act as a platform for the implementation of controlled printing.

Implementation of functional requirements for printing controlled documents and modifying user permissions (Req. 1, 2 and 3) can be done in the Web Access platform with a similar approach as in the previous solution. As Web Access supports the same features as the Windows desktop client, the controlled document objects, permissions and workflows can also viewed and edited in the browser environment. Web Access is not used solely on Windows environments, so the printing options might vary

based on the used browser and operating system. Access to the printing operation can be achieved with a controlled printing button that is added to the Web Access site.

The possibility to track printouts (Req. 4) can be achieved by modifying the new button that starts the printing operation. Every time this button is used to generate a printout, a new printout tracking object with timestamps and other necessary metadata is added to the vault.

Web Access does not have support for advanced printing operations such as adding labels to the document contents. Document watermarking (Req. 5) is difficult to implement without a service that can be used to modify the contents to include labels between the browser client and the server.

Tracking of controlled printouts (Req. 6) can be achieved by using built-in features of Web Access to send notifications. The document manager who attempts to track printouts can either track the state of printouts from these notification messages or manually search for the tracking objects that are created in the implementation of requirement 5.

Changing the default printing behavior (Req. 7) is implemented by default in Web Access as the controlled documents cannot be downloaded from the vault and then printed. This makes it possible to restrict the printout generation to function only with the controlled printing button that is implemented for requirements 1 and 4.

The strength of using the M-Files QMS Web Access as a solution for the implementation of controlled printing is that it can be accessed from many different environments, as long as the device and web browser are supported. Using the Windows desktop client does not allow for a similar level of multi-platform use which makes it difficult for users to create printouts without their desktop computers. Using a browser based application also makes it possible to use tablets and smartphones when attempting to create printouts.

The weakness of the solution is that Web Access is updated with new features later than the native application and as such the features that are first implemented to the Windows client take time to arrive to the web platform. This means that modifications done to the M-Files QMS desktop client can take up to six months until they are added to Web Access. Another weakness of Web Access is that there is currently no support for the user interface extensibility framework which makes it necessary to perform major changes to the Web Access core implementation in order to achieve all of the requirements.

4.1.3 M-Files QMS Extension with 3rd party frameworks

M-Files QMS offers a variety of possibilities for customization with its API and User Interface Extensibility Framework tools. The built-in features of M-Files QMS can be modified to include automated controlled printing on specific document objects in a way that requires little or no management by administrative users. 3rd party tools such as libraries that handle PDF file modifications can also be used.

The printing functionality (Req. 1) can be done by using the M-Files User Interface Extensibility Framework to add a controlled printing button to the task pane area of M-Files QMS. The button appears when a controlled document is selected and clicking it opens the native Windows printing dialog.

Preventing document printing from some users (Req. 2) is possible with the same functionality that is already used to implement requirement 1. In addition to making the controlled printing button to appear only for controlled document objects, printing permissions can also be verified as a part of the verification process.

Contrary to the solutions used to modify the authorizations (Req. 3) introduced in Subsections 4.1.1 and 4.1.2, the use of built-in permissions is replaced with a new list of authorized users. This list includes the users and user groups that can access the printing features. Viewing these permissions is possible from a new dialog that shows the list that is currently active and has controls for adding and removing users from it.

The requirement to track printouts (Req. 4) can be fulfilled by adding functionality to the controlled printing button. This new action adds a new printout tracking object automatically to the vault every time a printout is generated. The automation also adds the required metadata to these objects. In addition to this, the M-Files QMS server event log will show that the file was downloaded and printed.

Adding custom watermarks to printouts (Req. 5) can be done by accessing a 3rd party library that specializes in PDF content manipulation with the M-Files User Interface Extensibility Framework.

The traceability features (Req. 6) are available as built-in features in M-Files QMS. Notifications are sent to responsible users when the state of controlled document objects change. The same notification rule can be added to monitor the creation of printouts.

Setting default printing behavior (Req. 7) can be achieved by modifying the functionality of M-Files QMS so that the default Windows printing behavior cannot be accessed when selecting controlled document objects. As a result, the only possibility is to use the implemented controlled printing button.

The strength of adding new features to the M-Files QMS Windows client is that there is a large variety of tools available for programming and design purposes. Both the M-Files QMS API and User Interface Extensibility Framework can be used in conjunction with built-in features and 3rd party libraries that can be used to manipulate PDF files. All of the requirements can be implemented directly to the Windows client itself with minimal need for core modifications. This is important as most of the M-Files QMS customers are primarily using the native Windows client.

The weakness of using only the Windows client application as the basis for the implementation is that users who are not using Windows operating systems cannot access the controlled printing features. This restriction is crucial as Mac OS or Linux users cannot create printouts. Another weakness is that the use of 3rd party tools and programming libraries that are taken into use might also incur licensing fees or other service related costs on the implementation.

4.1.4 M-Files QMS with WatchDox integration

WatchDox offers possibilities for storing and sharing controlled documents from a cloud or on-premise service by means of a web interface. The controlled documents that are selected to be shared under the scope of the controlled printing feature can be transferred from the M-Files QMS vault to the WatchDox service storage by using the WatchDox API.

WatchDox has built-in support for secure printing operations which can be used to implement the generation of controlled printouts (Req. 1). The controlled documents can be transferred to WatchDox and then accessed from M-Files QMS with WatchDox API methods. A button needs to be added to the M-Files QMS client application to allow users to easily find and access the printing functionality. Clicking this button performs a check to verify that the controlled document can be printed, and then sends the printing command to WatchDox which then handles the printing job.

Preventing users from printing certain controlled documents (Req. 2) is possible by using the built-in security settings of WatchDox. The document manager logs in the WatchDox client service, selects the controlled document and disables the option to print the document for a list of users. Documents that are marked as restricted in WatchDox cannot be printed by these users in M-Files QMS. Metadata based security settings can also be set by means of automation using the WatchDox and M-Files QMS APIs.

Viewing and modifying existing security rules for controlled documents (Req. 3) can be done in the WatchDox client. A new browser dialog has to be implemented so that the printing permissions can be viewed and modified with the client's user interface. This new dialog has to list M-Files QMS users who have printing access to the document in question and also include buttons that allow the manager to add or remove users to the permission list.

Tracking of printouts (Req. 4) is possible by using WatchDox’s built-in ability to automatically store information relevant to the tracking requirements. This information includes a list of printouts, their creation timestamps and the names of the persons responsible for running the printing job.

WatchDox has support for adding pre-defined labels automatically to document printouts (Req. 5). These labels include default templates such as "draft" and "printed on date". Customized labels and watermarks based on the metadata stored in an M-Files QMS vault require that the WatchDox service is modified to have an access point to the target vault contents and metadata.

The required automation features (Req. 6) can be achieved with the WatchDox document tracking abilities. WatchDox has built-in support for sending notification messages to the document managers when user actions such as downloading, modification and printing are performed on controlled documents.

Due to M-Files QMS controlling the printing behavior instead of WatchDox, setting controlled printing as the default behavior for all printing operations (Req. 7)

requires an implementation that restricts access to the native Windows printing tools as previously explored in Subsection 4.1.3.

The strength of integrating M-Files QMS to WatchDox and then implementing controlled printing is that WatchDox offers a lot of the required features as built-in features. These existing features will save a lot of design and programming effort as advanced functions do not have to be created from the beginning. Another major strong point of WatchDox is that it allows storing of controlled documents and their tracking information on a cloud platform, enabling access to the server from domestic and international locations alike with only a small difference on the overall performance.

The weakness of using WatchDox as the basis is that licensing issues need to be taken into consideration. Licensing fees have to be negotiated with the supplier of the service and as a result might create additional expenses on the project. Also the documentation on the WatchDox API indicates that it aims only to make it possible to access WatchDox features such as labeling and printing from other applications and also implying that the API is lacking in features that allow 3rd party applications to modify the contents and functionality of the WatchDox service itself. This weakness can prove problematic as the implementation of controlled printing relies heavily on bidirectional API co-operation between M-Files QMS and WatchDox.

4.2 Copy Prevention

This section introduces two different solutions to the set of requirements for copy prevention of controlled documents. The implementation of copy prevention requires a number of additions and modifications to the M-Files QMS product that can only be solved with either core changes to M-Files QMS itself or delegating some of the work to a service or application that caters specifically to this purpose. Restricting the copying of controlled documents with modified M-Files QMS functionality is studied in Subsection 4.2.1. A solution that is based on integration with WatchDox's cloud-based security solutions is introduced in Subsection 4.2.2.

4.2.1 Modifying core functionality of M-Files QMS

There is currently no way to prevent an M-Files QMS user with read permissions from making a copy of a controlled document to their storage devices. M-Files QMS has been designed to interact seamlessly with other Windows applications and as such it allows users to read and edit document files as long as the M-Files QMS user has the necessary permissions. The downside of this is that if the user can read the file, he can also copy it outside of the vault in question. One solution to this problem is to modify the M-Files QMS client application to allow viewing the contents of controlled document only in the M-Files QMS window and also to take PDF documents with security features into use. If the user of M-Files QMS was successfully able to copy such a file to an external location, he would not be able to modify the contents or create printouts without knowing the password.

Preventing unauthorized copying of controlled documents outside of M-Files QMS (Req. 8) is possible with a server-side script that starts when files are being downloaded from the vault as a result of a copy-paste operation. The script checks if the downloaded file is a controlled document and then halts the operation effectively preventing any other ways of access to the documents than built-in preview function in the M-Files QMS client application. The previewer has no commands that allow the user to print or modify the document that is being accessed which makes the contents safe from tampering. The downside of showing the contents in any form in the previewer is that the text can still be copied with the Windows copy-paste tools. This is an acceptable weakness as the document's format, watermarks and style cannot be copied, making the copies easily distinguishable.

Adding an ability to create a downloadable PDF file that cannot be printed (Req. 9) is possible by accessing a 3rd party framework that specializes in modifying the security flags of PDF files. When a user wants to download a regulated version of a controlled document, the server uses the 3rd party framework to convert the source file to PDF format and enable the security flags. After the security alterations have been added to the new PDF copy, M-Files QMS server sends the file back to the user.

Preventing taking of screenshots (Req. 10) is possible by accessing the Windows API methods and using it to preventing the screenshot capture function from accessing the M-Files QMS document previewer, making the area look like a black box in the screenshot. This will require that changes are made to the core level features of M-Files QMS as the API does not allow for such complex operations on the operating system level. This screenshot restriction functionality will only include the M-Files QMS client application and does not extend to 3rd party software.

This approach to solving the copy prevention requirements has the strength of being a versatile option as all of the modifications are implemented directly to the M-Files QMS core functionality which makes it possible to prevent copying completely as file downloads can be halted before they even begin. Another strength is that the features that are created for the purpose of copy protection can be easily reused if any other M-Files product evolves to require a similar level of security. This solution can also be implemented completely in-house as the tools and know-how are readily available.

The weakness of this approach is that the simultaneous changes to integral M-Files QMS modules are complex to implement as these changes might affect a large variety of existing features and will require extensive testing. This use of additional employee resources has to be taken into account when planning the scope of the modifications. As this solution is also reliant on 3rd party frameworks, these will incur costs on the implementation as the required tools will likely include additional licensing terms and fees.

4.2.2 Copy prevention with WatchDox

One possible solution to achieve the copy prevention requirements is to transfer the responsibility of monitoring document security to an existing 3rd party service, one exam-

ple of such a service being the cloud-based WatchDox file sharing application. WatchDox includes built-in security features that protect stored controlled documents from being copied and modified by users that have no permissions to perform such actions. The controlled documents that require protection are moved from M-Files QMS to the cloud service and then regulated by using the WatchDox API.

Controlled documents that enter the state where they can no longer be modified are moved to WatchDox's cloud servers after which preventing copying of controlled documents (Req. 8) can be achieved by simply restricting editing and downloading access to these documents. As the watermarks and additional labels are already stored in the cloud server at the moment when they are copied there, they cannot be removed anymore without editing permissions. WatchDox offers a large variety of access points, but as the permissions are set on a user basis and the service is the same for all devices, it doesn't matter if the documents are accessed from a mobile device, web browser or a desktop PC. This makes WatchDox a secure and flexible option for copy prevention.

WatchDox has built-in features that allow copies of controlled documents to be downloaded on the device in use (Req. 9). Regulated copying can be allowed by enabling download permissions for the user in question and then having the user copy the controlled document from his WatchDox client application. These files have to be in PDF format and in a non-printable setting which will require that he controlled document is converted to PDF format by using 3rd party software as middleware or then using the WatchDox API to further customize the functionality [Watchdox, 2013C].

The screenshot capture prevention request (Req. 10) is supported by default in WatchDox. This function extends to all methods of taking screenshots on Windows desktop platforms and even supports masking the content from photographs taken of the screen [Watchdox, 2013D].

The strength of using a cloud-based client-server service is that it offers a large variety of powerful tools that can be accessed from multiple different devices. A cloud-based system also guarantees that stored information can be accessed from anywhere with minimal downtime. WatchDox offers a lot of different ways to handle copy prevention ranging from simple restrictions based on user permissions to advanced ones that are related to the location of the user. Implementing a prevention scheme on the same level of complexity will require a lot of work in other solutions.

The weakness of this approach is that it is complex to implement as changes have to be made into WatchDox and M-Files QMS applications simultaneously and the information transferred between them will require some level of conversion. The used 3rd party frameworks will incur additional costs on the project as the required tools will likely include licensing terms and fees. In addition to the license fees, the API tools are unknown to the employees of M-Files Corporation which means that some level of training is necessary.

4.3 Solution summary

This section summarizes the various solutions to the requirements, selects the most promising options, and then compares their approaches to the problems. The findings are collected to the tables that are found in the end of the section.

The solutions that were previously introduced for both controlled printing and copy prevention share options where the implementation is attempted with the built-in tools, with the in-depth modification of by the means of integration with WatchDox. As the requirements of both controlled printing and copy prevention are closely related to each other, the best possible solutions are those that accomplish all of the requirements and share a similar basis on the implementation. On this note, the first pair of solutions that have the same method of implementation for controlled printing and copy prevention are: using M-Files QMS extensions with 3rd party frameworks (Subsection 4.1.3) and modifying the core functionality of M-Files QMS (Subsection 4.2.1) and the second pair: integrating M-Files QMS with WatchDox for both controlled printing and copy prevention purposes (Subsections 4.1.4 and 4.2.2). The functional requirements for controlled printing and the solutions offered are listed and abbreviated in Table 4.1. The functional requirements for copy prevention and the methods of their implementation are shown in Table 4.2. Finally the non-functional requirements are listed compared for both the M-Files QMS modification and the WatchDox service in Table 4.3.

The controlled printing solutions that used only M-Files QMS features (Subsection 4.1.1) and Web Access (Subsection 4.1.2) are dropped from the comparison process as they are viable options only for making the controlled printing implementation possible. Both of these solutions are also missing functions that allow them to accomplish some of the requirements, notably in the area of adding watermarks to the printouts which is a requirement that has a high priority level.

The compared solutions offer similar approaches to most of the functional requirements set for controlled printing, the main difference being that the WatchDox implementation would need the document manager to access content and permissions from outside of the boundaries of M-Files QMS. This is apparent in the controlled printing requirements for modifying the user permissions and tracking the printouts (Req. 2, 3 and 4). The requested printout tracking and the related automation (Req. 4 and 6) features are a major strength even in the unmodified version of WatchDox and can be modified to offer solutions that are based on M-Files QMS metadata. On the other hand the requirement for watermarking (Req.5) is only possible by using default templates that have been included in the WatchDox service, and there is no straightforward way to add the important metadata based watermarking support that is easily implemented to M-Files QMS by using a 3rd party PDF manipulation framework. The methods that are used to modify the default printing behavior (Req. 7) are similar in both solutions.

The solutions that handle the copy prevention requirements are built-in in WatchDox, but the same level of protection is achieved with an adequate amount of work also in M-Files QMS. There is also a possibility that M-Files QMS customers will

protest against storing confidential information on a service that is not provided solely by M-Files Corporation. Both the creation of regulated PDF files (Req. 9) and prevention of downloads from an M-Files QMS vault (Req. 8) can be handled with simple server-side scripts. Preventing taking of screenshots (Req. 10) is a built-in feature of WatchDox, while in M-Files QMS it needs to be implemented as a core level extension. An important factor in deciding the copy prevention solution is should customer information be stored outside of the supplied M-Files QMS vaults and can the M-Files Corporation guarantee the security of the WatchDox service.

The total price for WatchDox (Req. 11) is calculated based on the amount of users, currently 15$ per named user per month, and the amount of optional features that are bundled into the service. An important factor is that the necessary WatchDox API framework has to be bought as one of these optional features [Watchdox, 2013A]. The price for implementation for the M-Files QMS modifications depends solely on if the 3rd party PDF manipulation tools, some of which are already included in the application, require any fees.

The Watchdox based solution's need for two separate client applications makes it difficult to develop a user friendly interface (Req. 12) as functions have to be divided to separate locations, likely causing the user to not finding the features intuitively. On the other hand the M-Files QMS modification based implementations would require access to only the M-Files QMS client application.

An adequate level of extensibility (Req. 13) for both M-Files QMS and WatchDox can be achieved by writing the application code meticulously following the M-Files Corporation programming guidelines. Both solutions have the potential for future growth and reutilization as long as the interfaces are designed and programmed correctly.

Both of the solutions are good options for the implementation of the requirements, but extending the built-in feature of M-Files QMS is the cheaper and more robust alternative as there is no need to rely on a large part of the application running on an independent 3rd party service. The deciding factors that weighed the decision are the requirements for an affordable price of implementation (Req. 11), the ability to customize the watermarks (Req. 5) and also the overall usability of the implementation (Req. 12). The user amount based licensing plan for WatchDox is unsustainable in cases where the customer has hundreds of unique users, especially when it is possible to modify M-Files QMS to handle all of the requirements with no additional monthly fees. The second major problem with WatchDox is that the user interface cannot be easily integrated to M-Files QMS, requiring the user to switch between two different user interface options that have nothing in common. This is a critical setback considering overall usability of the solution. The third problem arises from the WatchDox API which is an unknown factor, especially when considering the implementation of the M-Files QMS metadata based watermarking, when in comparison the requirement can definitely be achieved by using the M-Files QMS API together with 3rd party tools.

Table 4.1. *Comparison of Controlled Printing solutions.*

Functional requirement	**M-Files QMS extension with 3rd party frameworks**	**Controlled printing with WatchDox**
1. Possibility to create printouts from M-Files QMS	Add a button that opens the printing operations dialog to the M-Files QMS task pane.	Add a button to M-Files QMS that sends the document to WatchDox. WatchDox then handles the printing operation.
2. Possibility to exclude users from printing a document in M-Files QMS	The printing action checks user permissions and cancels the operation if it is not approved.	Manager logs in to WatchDox and allows selected M-Files users to print the document.
3. Possibility to view and edit current restriction rules	New permission list for controlled printing that can be modified from a specialized dialog.	New dialog within WatchDox that lists allowed users and allows the manager to add or remove them.
4. It must be possible to track every controlled printout	Creates an object that has log entry metadata to M-Files QMS after every printout.	WatchDox keeps a list of documents and printouts made of them.
5. Producing printouts with custom labeling	Use 3rd party PDF manipulation frameworks to add labels to printouts.	Possible to use pre-defined labels but customization requires API interaction with M-Files QMS.
6. Tracking controlled printouts with M-Files QMS notifications	Sends a notification mail message to responsible managers when printouts are created.	API interaction with the M-Files QMS e-mail notification tools.
7. Setting controlled printing as default printing behavior	Modify the M-Files QMS core so that Windows printing commands cannot be accessed.	Modify the M-Files QMS core so that Windows printing commands cannot be accessed.

Table 4.2. *Comparison chart for Copy Prevention solutions.*

Functional requirement	**Modifying the functionality of M-Files QMS**	**Copy prevention with WatchDox**
8. Preventing copying of controlled documents	Prevent the download of controlled document files	Restriction of read and edit permissions on the stored

	with a server-side script.	controlled documents.
9. Creating PDF documents that are not printable with standard viewers	The M-Files QMS server converts the document to PDF and disables modification and printing security flags.	No support for downloading documents as PDF files. Requires 3rd party framework support for PDF conversion.
10. Preventing taking of screenshots of controlled documents	Disable screenshots by modifying the M-Files QMS core.	Built-in support for screenshot prevention.

Table 4.3. *Comparison chart for non-functional requirements.*

Non-functional requirement	**Modification of M-Files QMS**	**WatchDox implementation**
11. Price of implementation	No additional costs as there are active subscriptions for necessary 3rd party tools.	Payment that is based on amount of users and optional features.
12. Usability	The user interface is integrated to the M-Files QMS client.	Offers a simple to use, multi-platform approach to user interface. On the downside, the interface is separated from M-Files QMS.
13. Extensibility	Further modifications can be implemented with M-Files API and User Interface Extensibility Framework.	Extensibility relies on the way WatchDox and its API have been implemented.

5 EVALUATION OF THE SOLUTION

This chapter explains the steps that are required to extend the functionalities of M-Files QMS to comply with the requirements for controlled printing and copy prevention. All of the requirements that were introduced in chapter 3 will be implemented to M-Files QMS by extending the client and server applications with the help of the M-Files API, M-Files User Extensibility Framework, and also a 3rd party framework which enables the manipulation of file contents programmatically.

The evaluation begins in Section 5.1 with the listing of multiple use cases that reflect common scenarios faced by users. These use cases help validate that the solution includes the tools to realize the requirements. Section 5.2 continues by listing the Aspose and M-Files Document Utilities components that are utilized in solving the custom watermarking request (Req. 5). Section 5.3 lists the functional and non-functional requirements and describes what kind of design decisions are needed to realize them. The evaluation is finalized in Section 5.4 with a showcase of the watermarking and printing functionalities.

5.1 Validation Use Cases

Use cases portray common user actions in situations that appear during normal use of the controlled printing and copy prevention features. The use cases are described accurately so that the portrayed actions can be reproduced by following the given conditions and steps. In the beginning of each case, the user and his role in the organization is introduced. The cases involve pre-conditions that need to be met before the described action can be replicated successfully. The end results and possible exceptions are also attached to the use case. The following use cases are based on the cases that are introduced in full in Appendices 1 and 2. The cases are used validate the controlled printing and copy prevention requirements against the solution, and verify that the M-Files QMS modification has the suitable functionalities to support common ways the implementation can be used by customers. Controlled printing requirements are covered in use cases 5.1, 5.2, 5.3, 5.4, and 5.5 while copy prevention requirements are included in use cases 5.6, 5.7, and 5.8.

Use Case 5.1. *Printing a controlled document.*

Actor:	M-Files QMS user
Pre-condition:	The M-Files QMS vault contains a controlled document that the user can read.
Action:	The user selects the controlled document object "Software Testing SOP" from the M-Files QMS objects listing and clicks the "Controlled printing" button that appears on the task pane area of the client window. The user selects the device that will handle the printing job from the printing dialog that opens and clicks "Print". The user collects the printed document from the device location, selects the created controlled printout object in the M-Files client and compares the printout to the stored copy of the original document. Finally the user moves the printout object's workflow to the "Successfully printed" state.
Exceptions:	**No printers connected:** A "Cannot find printer." error message is shown to the user, and the printing job is canceled. The printout object is still created and it moves to the state workflow "Printing failed". **No printing authorization:** A “Printing not authorized.” error message is shown to the user, and the printing dialog does not appear.
Results:	The printout is created successfully. A traceable printout object with identifying metadata is added to the vault, and a copy of the printed file is attached to it.

Use case 5.2. *Authorizing a user to print a controlled document.*

Actor:	Document Manager
Pre-condition:	The Document Manager has created the target document or has authorization to change its permissions.
Action:	The Document Manager selects the controlled document "Software Testing SOP". He then finds the “Authorized users” property definition and adds the user he wants to authorize to have the option to print the selected document. The Document Manager saves the changes that he made to the metadata by clicking the “Save” button on the metadata card.
Exceptions:	-
Results:	The user has been added to the authorized printers list, and can now print the "Software Testing SOP" document.

Use Case 5.3. *Adding labels and watermarks to a controlled document.*

Actor:	Document Manager
Pre-condition:	The Document Manager has created the document or has authorization to edit the controlled document.
Action:	The Document Manager creates a new style template object and names it "SOP style template. He then selects the newly created object in the object listing and adds two watermark objects "SOP Footer" and "SOP Header" to which he sets the anchor property values "bottom" and "top", and selects the displayed metadata as object ID. He sets both their position properties as "100;20" (x;y). Finally he saves the changes. The Document Manager searches for stored controlled document objects and selects the document "Testing environment SOP". He applies the "SOP style template" as its watermark template property value and saves the changes.
Exceptions:	-
Results:	Further printouts produced of the controlled document will include the header and footer watermarks.

Use Case 5.4. *Tracking the state of printouts made of a controlled document.*

Actor:	M-Files QMS user
Pre-condition:	There are printout objects in the vault to which the M-Files QMS user has read access to.
Action:	The user selects the controlled document "Software Testing SOP" whose printouts he wants to locate. The user expands the related objects of the controlled document in the view and counts how many printout objects are attached to the controlled document. The user browses through all of the printout objects and studies the "Printout ID", "Printing Date" and "Recipients" properties.
Exceptions:	-
Results:	The user has found out how many printouts of the controlled document have been created, their versions, and the recipients of the printouts.

Use Case 5.5. *Recalling a controlled document printout.*

Actor:	Document Manager
Pre-condition:	There are printout objects that are in the "Successfully printed." workflow state.
Action:	The document manager selects the controlled document "Software Testing SOP" and then expands the document's related objects. The manager selects the printout "PrintID1: Testing environment SOP" and moves its workflow state to "Recall issued". A notification e-mail about the recall is sent to the user who printed the controlled document. The manager receives the printout that was recalled and changes the printout state to "Recall successful".
Exceptions:	**The printout has been lost:** The manager sets the printout's workflow state as "Recall delayed" until the printout has been found.
Results:	The document manager has successfully recalled the obsolete printout from circulation.

Use Case 5.6. *Attempting to open a controlled document outside M-Files QMS.*

Actor:	M-Files QMS user
Pre-condition:	There are controlled documents in the vault.
Action:	The user starts Adobe Reader and browses to the "QMS" vault. He searches for the controlled document "Software Testing SOP" and attempts to open it in the reader application. The user is met with a "File cannot be opened." error message which prevents the viewing of the document.
Exceptions:	**The user is an administrator:** The user can download the controlled document regardless of the copy prevention settings.
Results:	The user cannot open the controlled document outside of M-Files QMS.

Use Case 5.7. *Downloading a regulated version of a controlled document.*

Actor:	M-Files QMS user
Pre-condition:	There are controlled document objects in the vault to which the M-Files QMS user has read access to.
Action:	The user selects a controlled document that is listed in the vault. He clicks the "Download controlled document" button on the left side task pane. M-Files QMS prompts the user to select a target folder which he selects as "C:\Documents\". Finally the user clicks the "OK" button.
Exceptions:	-

Results:	The user has successfully downloaded a read-only copy of the controlled document.

Use Case 5.8. *Taking a screenshot of a controlled document's content.*

Actor:	M-Files QMS user
Pre-condition:	There are controlled document objects in the vault to which the M-Files QMS user has read access to.
Action:	The user selects a controlled document and then clicks the preview tab on the right side pane. This opens the file in preview mode. He then proceeds to take a screenshot by pressing the "Print Screen" key. The user pastes the screenshot to the Paint application.
Exceptions:	-
Results:	The screenshot shows a blank area in place of the controlled document preview pane.

5.2 Utilized software components

The project requires the use of a 3rd party framework and in-house tools developed for the purpose of making developing of M-Files extension modules easier. The utilized tools include File Format Components developed by Aspose Pty Ltd, and the M-Files Document Utilities Framework. The features of these tools are explained in Subsections 5.4.1 and 5.4.2.

5.2.1 Aspose – File Format Components and Controls

The Aspose File Format Components and Controls allow programmers to manipulate the contents of many different file types. The support list of the Aspose components includes commonly used file types such as PDF, JPG, PNG, Office documents and XML. Even though the structures of the different file types vary dramatically, Aspose declares interfaces that hide the underlying methods under similar method calls. These methods handle low level operations such as adding a line of text or a picture to a file. The framework has been previously used in the M-Files QMS server application to convert files to PDF format either after a document has been scanned in a picture format or when a user wants to store a document file in a format that has multi-platform support. The components that are required in the scope of implementing the necessary modifications include the ones that are able to manipulate PDF, XML and Office documents. This list of supported file types can be expanded in the future based on possible customer requests. The Aspose tools can be accessed from the M-Files QMS server application's scripting tools when the need for manipulating documents arises [Aspose, 2014].

5.2.2 M-Files Document Utilities Framework

Modifying file contents by using only the built-in Aspose interfaces can be a demanding task for programmers who do not have earlier experience with the components. The M-Files Document Utilities Framework encapsulates complex operations inside classes and methods for the purpose of allowing programmers to effortlessly manipulate files that have been stored in an M-Files vault. If these same operations were implemented with M-Files API and Aspose scripting, this would result in code that is complex and difficult to maintain. In the scope of the controlled printing and copy prevention implementation, required operations include converting files to PDF format, adding simple and metadata based watermarks to files and also the modification of PDF security settings. The M-Files Document Utilities Framework runs as an optional service which can be accessed from the M-Files server's scripting tools and also .NET Framework based applications.

5.3 Technical implementation of requirements

The aim of this Section is to explain the fulfillment of the functional and non-functional requirements in a technical manner in order to validate the claims that were set earlier in Subsections 4.1.3 and 4.2.1. The necessary M-Files QMS metadata structure modifications, required scripting support and also the used 3rd party tools are also listed and explained in the following Subsections 5.2.1 and 5.2.2. The requirements are prioritized based on the order of their importance, and the necessary resources are allocated based on this sorting when the actual implementation project begins.

5.3.1 Functional requirements

This subsection introduces the technical aspects to implementing the functional requirements for controlled printing and copy prevention. The explanations include steps to achieve the desired result and also the tools that have to be utilized in order to make these steps possible.

1. Possibility to create printouts from M-Files. The printing operation is started by clicking a button that has been added to the M-Files QMS' client application task pane with the User Interface Extensibility Framework. The button is hidden until the user selects an object that is based on the "Controlled Document" object type". The button is also hidden for controlled documents that have been checked-out for modification by another user, and also if the user has selected multiple controlled documents simultaneously. Clicking the button invokes an event in the server-side application which converts the original controlled document to PDF format, and also updates the content to include watermarks and labels (Req. 5). This PDF file is returned to the client application after the modification, and a printing dialog is shown to the user. The user can select the used printer device from this dialog and finally initiate the printing job.

2. Possibility to exclude users from printing a document in M-Files. The ability to prevent unauthorized users from printing controlled documents is added by extending the implementation of requirement 1. The document manager user adds the authorized users to a user group called “Authorized for controlled printing”. When the controlled print button is clicked the printing operation checks if the currently logged-in user belongs to this user group. The operation is allowed if user can be found in the group, and canceled if this is not the case. The document managers can also restrict unauthorized printing by removing the read permissions to the controlled document object from users that are not allowed to create printouts.

3. Possibility of viewing and editing current restriction rules. The list of authorized users implemented in requirement 2 can be viewed in a dialog that is added as a User Interface Extensibility application to the M-Files QMS client application. The document manager users who are responsible for the controlled documents have an “Edit controlled printing authorizations” button visible in the task pane area of the client application. Clicking this button opens a dialog that lists the authorized users. The document manager can then either add new M-Files QMS users to it, or remove existing ones from it.

4. It must be possible to track every controlled printout. The tracking of printouts is made possible by an automatic action that creates printout related tracking objects every time a new printout is generated from a controlled document. These tracking objects are based on a new custom object type ‘Controlled Printout’ which contains required property definitions for identifiers, effective dates, printing dates, and responsible users. The action also attaches a copy of the printed and watermarked version of the controlled document to the object. This method guarantees that all printouts have a digital footprint in the vault and that they are also traceable. The unique tracking objects can be found from the vault either by a direct reference from the controlled document, or by running a search operation.

5. Producing printouts with custom labeling. Adding custom watermarks to the printouts is possible by manipulating the source document programmatically with the help of the M-Files Document Utilities and Aspose Frameworks. The rules that state how watermarks are applied to files are stored in the vault as overlay template objects, which contain specialized watermark sub-objects for images and text fields. These overlays can be referenced by multiple separate controlled document objects, enabling the creation of complex and reusable watermarking rules. The controlled printing application interprets these referenced overlay templates when the controlled printing button is clicked, and then applies their contents to the printouts in the server-side application with the help of M-Files Document Utilities’ watermarking classes and methods.

6. Tracking of controlled printouts with M-Files QMS notifications. M-Files QMS includes built-in automation features that allow sending of e-mails to one or multiple recipients. These e-mail notifications can be sent when certain situations such as object creation or workflow state change occurs. In the case of the tracking of printouts, the e-mails are sent every time a printout tracking object has been created or when a tracked printouts state has been modified. The users on the mailing list are the owner of the printout and all other users who have been listed as the recipients. The e-mail notifications help document managers track changes as the messages contain metadata such as timestamps, modification descriptions and names of responsible users.

7. Setting controlled printing as default printing behavior. Setting the controlled printing action to replace the default windows printing dialog in M-Files QMS is possible by using the M-Files User Interface Extensibility Framework to hide the printing option from all menus while a controlled document object is selected by the user. This way the Windows printing dialog cannot be accessed by the user, and the only remaining way to run a printing operation is by using the controlled printing button.

8. Preventing copying of controlled documents. Preventing copying of controlled documents in M-Files QMS can be accomplished writing a script to the server application's file download event handlers. This script identifies the object from which the file is being downloaded from and then prevents the file download from proceeding if it is a controlled document. This prevention makes it effectively impossible to download controlled documents from the server.

9. Creating PDF documents that are not printable with standard viewers. Creating the requested PDF files that cannot be printed or edited can be done on the server application with the help of the M-Files Document Utilities Framework which can alter the security states of PDF files. A new button is added to the client application's task pane area and it is used to start the download of a secured PDF file. This PDF file can only be downloaded by users that are included in the authorized users list that was implemented for requirement 2.

10. Preventing taking of screenshots of controlled documents. Modifying the functionality of M-Files QMS client application to hide the content of viewed controlled documents is achievable by changing M-Files QMS core programming so that the previewer area is not included in the taken screenshots. This is possible by intercepting the Windows API call that requests the client window to return its state to the screenshot function and this way preventing the screenshot from showing the file preview area of the client. This will result in the produced picture showing only a black area instead of the source content. The prevention will only function when content is being viewed with the M-Files QMS client application and will not extend to software used to view the contents of downloaded PDF files.

5.3.2 Non-functional requirements

This subsection explains how the non-functional requirements of controlled printing and copy prevention are achieved with the evaluated solution. The explanations include steps to achieve the desired result and also the tools that have to be utilized in order to make these steps possible.

11. Price of implementation. The price of implementation is limited to the licensing fees of the required 3rd party frameworks. In the scope of these requirements the only framework that is needed is the Aspose File Format Components that are already included in the M-Files QMS core product. With the development tools and licenses already in place, there is no need to pay additional fees for the implementation of controlled printing and copy prevention.

12. Usability. The usability requirements of the implementation are achieved by modifying strictly the M-Files QMS client application's user interface and making the changes to the existing interface minimal. In addition to keeping the implementation's interface as similar to the client application as possible, the client application has to stay responsive even in situations where the printed documents files are large. M-Files QMS has been approved by extensive usability testing, which means that a desirable level of usability can be achieved as long as only non-intrusive changes are made to the user interface. The only visible modifications that are added from the client users' perspective are the controlled printing button and secured PDF download buttons on the task pane. The utilized dialogs are either based in existing M-Files QMS functionality, or are accessible from the Microsoft .NET Framework.

13. Extensibility. The requirement for the implementation's extensibility can be achieved by designing the interfaces, classes, methods and attributes in a generic manner that allows them to be used also in other M-Files products and upgraded to meet new customer requests. All of the frameworks and tools utilized in the implementation are designed for object-oriented development, which makes it straightforward to encapsulate the functions as separate components. The resulting components are based around each task in the overall printing workflow. These tasks are the printout generation, watermarking, the user interface elements, PDF security modification and tracking automation. The encapsulated nature enables carrying out modifications with a minimal risk of the changes affecting other than strictly related components.

5.4 Implementation sample: Watermarking and Printing

This section offers insight to the implementation of controlled printing and copy prevention from the perspective of the application of watermarks to a controlled document after a printout has been requested by an M-Files QMS user. The showcase is based on the requirement having the ability to print a controlled document (Req. 1), for adding

custom labeling to printouts (Req. 5) and also indirectly related to the requirement for tracking and recalling printouts after their publication (Req. 4). This is due to the fact that tracking relies heavily on labels such as unique identification numbers and printing dates. This sample was designed and programmed for the M-Files Corporation in order to test that the solution can be implemented in practice. The architecture of the watermarking implementation is introduced in Subsection 5.4.1, after which Subsection 5.4.2 explains the required vault metadata structure modifications, and finally the steps to applying a watermarking overlay on a printed file are shown in Subsection 5.4.3.

5.4.1 Architecture

The showcase implementation is based on the client-server architecture; in this case consisting of the M-Files QMS client and server applications. Both of these applications are extended to contain multiple components that specialize in different tasks required to accomplish the watermarking and printing operation that begins when the user selects a controlled document and clicks the controlled printing button.

The client side section of the implementation consists of two different components which are the .NET based client operations service and the User Interface Extensibility Application. The role of the client operations component is to collect information required to proceed with the printing operation, and then communicate with the server interfaces. It also utilizes the .NET Framework when there is a need to interact with printer devices. The User Interface Extensibility Application component is used to implement the controlled printing button to the M-Files client application task pane, verify user authorization, and also validate that the selected document matches required printing parameters such as supported file types.

The server-side section consists of components that are required to download files from the M-Files QMS vault and manipulate their contents. The server operations component implements the interface that is used to call the server application from the client operations component. The interface is also used to transfer necessary parameters such as the identifier of the selected controlled document. The server operations component fetches the file and the watermarking objects from the M-Files QMS database and transfers them to the watermarking component. This component then interprets the watermarking related values such as position, rotation, and font, sending them forward to the M-Files Document Utilities and Aspose components. These components are responsible for the actual application of the watermarks to the target file. The overall structure of the architecture and the connections between the introduced components are shown in Figure 5.1.

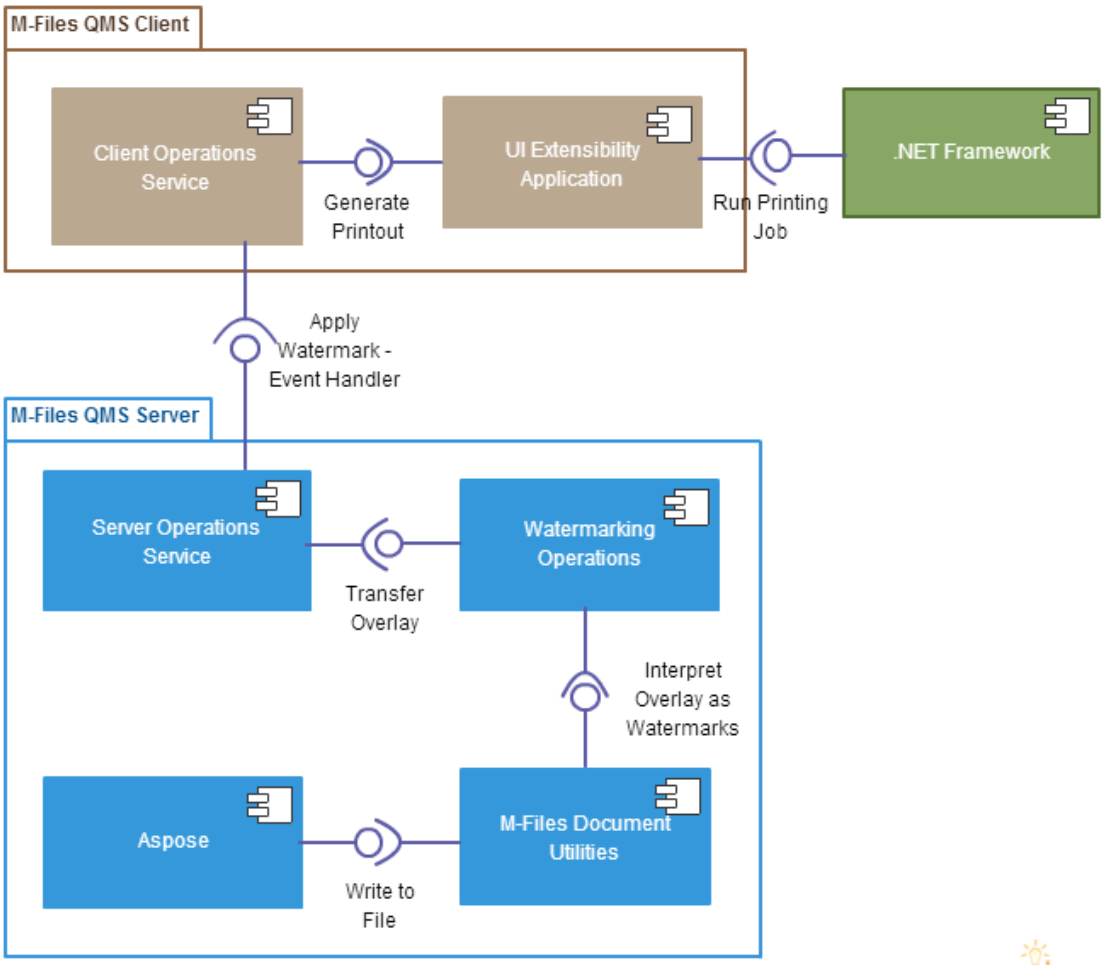

Figure 5.1. *Architecture of the Watermarking implementation.*

5.4.2 Vault metadata structure

The metadata structure of the M-Files QMS vault is modified to support the implementation of watermarking the generated printouts. This is done by creating four new object types: Controlled Document, Controlled Printout, Overlay Template and Watermark. Controlled Printout objects are created to contain copies of the files that are originally attached to the Controlled Document objects in order to track the state of printouts. The object types contain a two-way reference which can be navigated in order to list the source document of a printout and also all of the printouts that have been generated of the Controlled Documents.

Watermarking related metadata is stored in Overlay Template objects which are composed of a variety of Watermarks. An Overlay Template object can, for example, be formed of a header watermark, a footer watermark, a company logo stamp, and metadata based Watermark objects. The object is attached to controlled documents object as a reference which is then used to navigate to the Watermark objects when the watermarking operation begins. Other metadata structure changes consist of new property definitions that are used to identify and describe the new object types.

The metadata structure is represented by the class diagram shown in Figure 5.2. Object types are represented by classes, their relations as connecting lines, and the new property types are shown as class attributes. All of these metadata structure changes can be implemented manually in the M-Files QMS Server Administration application. Further descriptions of the property definitions that are attached to the object types are given in Table 5.1, 5.2, 5.3, and 5.4.

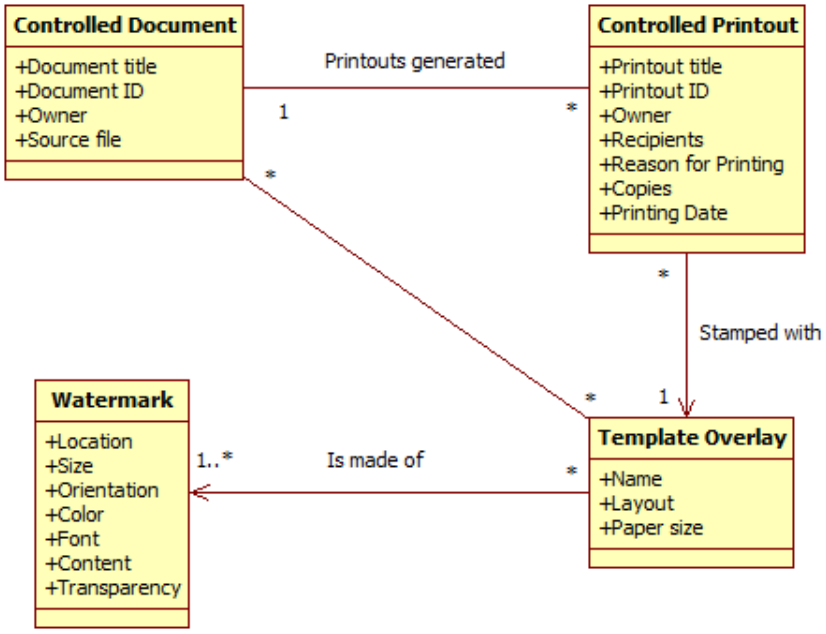

Figure 5.2. *Class diagram representation of the Metadata Structure.*

Table 5.1. *Property definitions of the Controlled Document object type.*

Property definition	**Type**	**Description**
Document title	Text	The name of the controlled document.
Document ID	Text	A unique controlled document identification number.
Owner	Object reference	A reference to the user who manages the controlled document.
Source file	Object reference	A reference to the controlled document file.
Overlay	Object references	Reference to used template overlay objects.

Table 5.2. *Property definitions of the Controlled Printout object type.*

Property definition	**Type**	**Description**
Printout title	Text	A unique title which includes the name of the source

		document, amount of copies and the unique printout ID.
Printout ID	Numeric	A unique identification number for the printout.
Owner	Object reference	A reference to the user who printed the document.
Recipients	Object references	References to all of the users and user groups that are the recipients of the printout.
Reason for printing	Multi-line text	A written comment stating the reason for the printing.
Copies	Numeric	The total amount of copies that have been printed.
Printing Date	Date	The date on which the document was printed.
Source Document	Object reference	A reference to the source controlled document.

Table 5.3. *Property definitions of the Template Overlay object type.*

Property definition	**Type**	**Description**
Name	Text	Name of the overlay template.
Layout	Text	The layout orientation, for example: Portrait or Land-scape.
Paper size	Text	The paper size of the overlay, for example: A4 or A3.
Watermarks	Object references	References the watermarks that are included within the overlay template.

Table 5.4. *Property definitions of the Watermark object type.*

Property definition	**Type**	**Description**
Location	Text	Location data stored as X and Y coordinates.
Size	Text	Size data stored as width and height.
Orientation	Numeric	Orientation data stored as degrees.
Color	Text	The watermark color in hexadecimal format, for exam-ple #FFFFFF (white).
Font	Text	The name of the used font.
Content	Text	The text that is displayed in the watermark.
Transparency	Numeric	A transparency percentage ranging between the values 0-100.

5.4.3 Printing a controlled document with a watermark overlay

The aim of the showcase implementation is to enable an action that includes the watermarking of a controlled document and then printing a copy of it with a selected printer device. As a pre-condition, the vault where this action is performed needs to have controlled documents, overlay templates, and watermark objects stored inside it. The selected controlled document object must have at least one overlay object reference in order to enable the watermarking. The overall printing and watermarking operation is visualized in the flowchart show in Figure 5.3.

The printing and watermarking operation starts when an authorized user clicks the controlled printing button that appears on the task pane area after a controlled document objects is selected. Clicking this button calls a printout generation method in the client operations component. This method validates that the type of the file stored in the controlled document is one that the Aspose component can manipulate and then fetches the identifier and latest version of the currently selected object.

The client operations component sends a file modification request to the server by invoking an event handler that the M-Files QMS server application provides in order to begin the actual watermarking process. The event handler is given the target document's identifier and version number as parameter. The server operations component uses these parameters to locate the selected controlled document object, referenced overlay templates, and the related file from the vault database. After these have been found, the component relays their locations to the watermarking component as references.

The watermarking component fetches all of the watermark objects that have been attached to the overlay objects and then accesses their property values in order to collect data that is necessary for the watermarking to begin. This necessary data includes the values that are shown in Table 8. The component sends these values to the Document Utilities encapsulated watermarking method which starts adding the watermarks to the file one at a time by calling the Aspose components file manipulation methods. After all of the watermarks have been applied, the file is converted to PDF format.

The server finalizes the operation by creating a new printout object to the vault, attaching the newly watermarked file to it, and sending a reference to it back to the client operations component. The component receives the file and starts the printing job by calling the printing library of .NET Framework. This last step opens a printing dialog to the user who then selects a printer device and properties such as the color scheme and amount of copies.

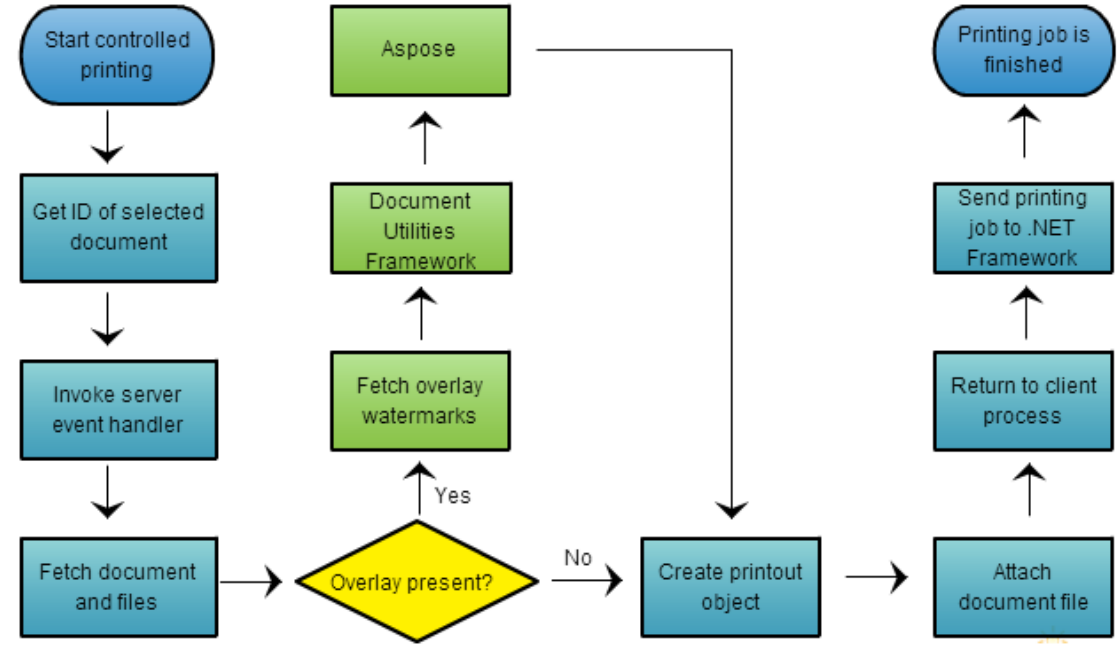

Figure 5.3. *Flowchart of the watermarking and printing operation.*

6 CONCLUSIONS

This thesis examined the possibility of extending the existing document control features of the M-Files QMS product to include a way of printing traceable documents, manage the lifetime of documents in circulation, and regulate sharing of documents outside of organizational boundaries. Potential solutions were weighed against a set of requirements that were collected from both internal requirements and customer requests. One of these solutions was picked for further evaluation based on the overall accomplishment of both functional and non-functional requirements. This solution was the extension of the M-Files QMS product to include support for frameworks and new functionalities that act as the basis for controlled printing and copy prevention implementation.

The evaluation of the candidate solution indicates that all of the assigned functional and non-functional requirements can be accomplished by utilizing M-Files QMS client and server applications that have been extended by use of M-Files API, M-Files User Interface Extensibility Framework, Aspose components and the M-Files Document Utilities Framework. This observation was backed by use cases that were used to validate high priority requirements and also by an implementation sample which demonstrated M-Files QMS's ability to communicate with these 3rd party components and store process related data in its metadata structure. As the showcase already contains the design and basic implementation for some of the high-priority requirements (Req. 1, Req. 2, Req. 3, Req. 4, and Req. 5) which were possible with minor scripting and metadata modification efforts, the rest of the requirements should also be feasible by following the designed solution. The remaining requirements are added in the actual implementation project which has been included in the M-Files QMS 3.0 iteration development plan.

The addition of the remaining requirements should be sorted into two categories: ones that can be easily added by modifying vault metadata and ones that require moderate amounts of work as they require core modifications and scripting. The requirements that fall into the first category are the verification of printing permissions (Req. 2), modification of printing permissions (Req. 3), creation of log entries for tracking purposes (Req. 4), and tracking printouts with notifications (Req. 6). The second category includes the creation of printouts (Req. 1), application of watermarks (Req. 5), replacing the default printing behavior (Req. 7), preventing copying of documents (Req. 8), creating secured PDF files (Req. 9), and finally preventing the taking of screenshots (Req. 10). The implementation should be started from the high-priority requirements that belong to the first category, and then continued with the second category options after the groundwork for the controlled printing and copy prevention features has been built. This approach allows for quick results to be made in the beginning, and for the developers to

get used to the necessary tools and environments. The more difficult requirements can be developed later, after a level of familiarity is reached.

The added functionalities can likely be used in other M-Files products as the need for copy prevention and document tracking are present in all document management applications, only in a less regulated manner. One example of a future upgrade that would take the watermarking functionality into use would apply watermarks to record files that are stored in the M-Files QMS vault. If a future project involves modifying the implementation to one of these other applications, the non-functional requirements for cost, usability and extensibility are also of major importance.

Even though WatchDox was not selected as the approach to the implementation in this scope, in the future it might still be taken into use as the next step in extending the functionalities of the Copy Prevention's sharing functions. The requirements in this thesis included simple cases where the user needs to create regulated PDF copies of controlled documents locally. Limiting the scope of the implementation was a good choice considering the simplicity of these requirements, but in the future a need might arise for a way to share controlled documents externally. These new use cases include situations where an organization has offices in multiple locations, or it has a need to share quality documents with subcontractors. The copy prevention, monitoring and tracking support that WatchDox provides could be a vital asset in this kind of an implementation.

REFERENCES

Adobe Systems Incorporated. 2013A. About Adobe PDF. [WWW]. [Accessed on 14.10.2013]. Available at: http://www.adobe.com/products/acrobat/adobepdf.html

Adobe Systems Incorporated. 2013B. Creating an interactive PDF file from a multipage document in Fireworks CS4. [WWW]. [Accessed on 15.10.2013]. Available at: http://www.adobe.com/devnet/fireworks/articles/interactive_pdf.html

Aspose. 2014. Aspose.Total for .NET. [WWW]. [Accessed on 15.01.2014]. Available at: http://www.aspose.com/.net/total-component.aspx.

Autodesk Incorporated. 2013. How to Remove PDF Password Security (Print/Edit/Copy) In Three Easy Steps. [WWW]. [Accessed on 22.12.2013]. Available at: http://www.instructables.com/id/How-to-Remove-PDF-Password-Security-PrintEditCo/

Aveta Solutions. 2013. Why Quality Management is Important. [WWW]. [Accessed on 10.9.2013]. Available at: http://www.sixsigmaonline.org/six-sigma-training-certification-information/why-quality-management-is-important.html

Cachadinha, N. 2009. Implementing Quality Management Systems in Small and Medium Construction Companies: A Contribution to a Road Map for Success. Leadership Manage, Volume 9, Issue 1. p. 32-39.

Hoyle, D. 2009. ISO 9000 Quality System Handbook. 6. Edition. USA, Taylor & Francis. 824p.

International Organization for Standardization. 2012. Quality management principles. [WWW]. [Accessed on 3.9.2013]. Available at: http://www.iso.org/iso/qmp_2012.pdf

Microsoft. 2013. Information Rights Management in Office 2010. [WWW]. [Accessed on 13.10.2013]. Available at: http://office.microsoft.com/en-001/excel-help/information-rights-management-in-office-2010-HA010354260.aspx

M-Files Corporation. 2012A. M-Files Software Architecture. [Accessed on 20.10.2013]. Available at: M-Files Corporation internal document vault.

M-Files Corporation. 2012B. M-Files UI Extensibility Framework. [WWW]. [Accessed on 15.10.2013]. Available at: http://www.m-files.com/UI_Extensibility_Framework/webframe.html

M-Files Corporation. 2012C. Guide to the Controlled Document Workflow. [Accessed on 20.11.2013]. Available at: M-Files Corporation internal document vault.

M-Files Corporation. 2012D. M-Files QMS Implementation Guide for the System Owner. [Accessed on 20.11.2013]. Available at: M-Files Corporation internal document vault.

M-Files Corporation. 2013A. M-Files. [WWW]. [Accessed on 9.8.2013]. Available at: http://www.m-files.com/

M-Files Corporation. 2013B. M-Files Help Manual. M-Files 10.

M-Files Corporation. 2013C. M-Files API Manual. M-Files 10.

M-Files Corporation. 2013D. Records Management feature requests to R&D from PM. Available at: M-Files Corporation internal document vault.

M-Files Corporation 2013E. QMS 2.0 - Document Control. Available at: M-Files Corporation internal document vault.

Quality Systems Toolbox. 2013. Control of Documents. [WWW]. [Accessed on 15.10.2013]. Available at: http://www.qualitysystems.com/support/pages/control-of-documents

Rumane, A. 2010. Quality Management in Construction Projects. 1. Edition. USA, CRC Press. 464p.

TechCrunch. 2009. WatchDox Lets You Track And Control Document Sharing. [WWW]. [Accessed on 15.10.2013]. Available at: http://techcrunch.com/2009/06/01/watchdox-lets-you-track-and-control-document-sharing-beta-invites/

The 9000 Store. 2013A. What is ISO 9001 Quality Management System? [WWW]. [Accessed on 10.9.2013]. Available at: http://the9000store.com/what-is-iso-9001-quality-management-system.aspx

The 9000 Store. 2013B. What is a Process Approach? [WWW]. [Accessed on 17.9.2013]. Available at: http://the9000store.com/ISO-9000-Tips-Process-Approach.aspx

The 9000 Store. 2013C. ISO 9001:2008 Documentation Requirements. [WWW]. [Accessed on 17.9.2013]. Available at: http://the9000store.com/ISO-9000-Tips-Documentation-Requirements.aspx

Verkamo, T. 2004. Ohjelmistotuotanto, vaatimusanalyysi. [WWW]. [Accessed on 25.6.2013]. Available at: http://www.cs.helsinki.fi/u/taina/ohtu/s-2004/luku4.pdf

Watchdox. 2013A. About WatchDox. [WWW]. [Accessed on 15.10.2013]. Available at: http://www2.watchdox.com

Watchdox. 2013B. Secure File Sharing & Sync. [WWW]. [Accessed on 15.10.2013]. Available at: http://www2.watchdox.com/wp/solutions/secure-file-sharing-sync-2/

Watchdox. 2013C. Knowledge Base. [WWW]. [Accessed on 08.11.2013]. Available at: http://www2.watchdox.com/help/when-i-am-accessing-a-document-through-the-watchdox-web-application-i-am-prompted-to-download-it-instead-of-being-able-to-view-it-online/

Watchdox. 2013D. Can you protect a document from screen capture? [WWW]. [Accessed on 08.11.2013]. Available at: http://www2.watchdox.com/help/can-you-protect-a-document-from-screen-capture/

APPENDICES

Appendix 1: Customer requirements

Preventing uncontrolled printing and copying

1. It must be possible to prevent printing of a document from M-Files
2. Prevention must be selective
    a. by class, metadata or state (e.g. effective SOPs cannot be printed)
    b. by user or group (e.g. SOPs can only be printed by members of Quality managers group)
    c. any combination of the two
3. System should prevent copying selected documents as flat files outside the M-Files system, as those documents may be printed without any control
4. Setting up printing restrictions should be relatively simple task, and accomplished by standard trained M-Files administrator
5. It should be possible to find out what the printing restrictions are for a given class, given user/groups, or system wide
6. It should be possible to produce PDF documents that are not printable with standard PDF viewers i.e. the controls against printing is embedded in the file itself

Producing printouts with controlled labeling

7. It must be possible to generate printouts with labels consisting of
    a. static text items (e.g. “UNCONTROLLED COPY” as header, footer, sidebar or watermark)
    b. metadata items (e.g. date of printing)
    c. stamps or images
8. Producing printouts with controlled labeling should be easy, and require no special skills or actions from user

9. Printout with controlled labeling should be the default print behavior for selected document classes
10. It should be possible to capture the default print commands of Windows, M-Files client or web browser, so every printout would be labeled in a controlled fashion
11. It should be possible to capture Windows Ctrl-P shortcut to produce a controlled printout for selected classes

Keeping log record of controlled document printouts

12. It must be possible to track every controlled printout within M-Files either via:
    a. system log entries i.e. audit trail (not preferred, but acceptable)
    b. dedicated M-Files objects or documents (preferred)
    c. both (the best option)
13. Track entry of each controlled printout must contain:
    a. user who printed
    b. timestamp
    c. the exact document and its version (ID, major, minor and M-Files built-in version numbers)
    d. unique numeric ID of the printout (preferably item-specific running number i.e. if 10 printouts are made of a controlled document, the printout copies or objects should be numbered 1…10, instead of running number across all printings)
14. The track of controlled printouts should enable normal available methods for M-Files automation, notification and alerts

Example SOP Use Case:

- A new SOP (that is: version 1.0) becomes effective, is automatically labeled as read-only record in the approval workflow, and is published within the company via necessary means
- Paper copies are also needed in the factory; production manager opens M-Files and produces 10 controlled printouts, each with necessary labeling, date of printing, and running number of each printout copy
- Manager places the printouts in binders near where they are needed
- Some users view the SOP on computer screen, but are unable to produce ad hoc printouts
- 6 months later the SOP is updated to version 1.1 which replaces 1.0
- Document manager will check if there are printouts of the previous version in use. As records about such copies are found they are assigned to responsible person for destruction. Documents are then manually collected, destroyed, and related assignments in M-Files closed.
- Additionally, document managers group may be notified via email in case there are incorrect printouts i.e. controlled prints of obsolete versions in use
- At any moment M-Files will provide, through views, reports or filtering of the system log, a full listing of all controlled printing activities, including destruction of obsolete printouts

Appendix 2: Customer requirements sample

Requirement ID	Description
URS 5.17.01	The system must provide the ability to record the name of the individual who printed a document.
URS 5.17.02	The system must provide the ability to support uncontrolled printing of documents. Uncontrolled printing is the ability to print a document without capturing any information of when or who printed the document. The need for controlled printing shall be based on Document Type.
URS 5.17.03	The system must provide the ability to support issued printing of documents. Issued printing is the generation of individually numbered prints of master documents.
URS 5.17.04	The system shall provide the ability for an administrator to configure the users and groups that are authorized to print and/or reprint controlled copies.
URS 5.17.05	The system shall provide the ability for an administrator to configure which lifecycle status to allow users to print uncontrolled copies of a document from the read-only rendition.
URS 5.17.06	The system shall provide the ability for an administrator to configure users to print controlled copies of a document for a configured lifecycle status.
URS 5.17.07	The system must provide the ability to support the definition of printing parameters including the page ranges printed and recipients of printed copies for controlled printing.
URS 5.17.08	The system must provide the ability to control which printer devices (hardware) may be used to print controlled copies.
URS 5.17.09	The system must provide the ability to ensure that the controlled print copy numbers generated are universally unique numbers for each version of the document. For example, that two people printing at the same time receive unique numbers.
URS 5.17.10	The system shall provide the ability for an administrator to configure whether recipient names must be chosen from a list, entered as free text, or a combination.
URS 5.17.11	The system shall provide the ability for an administrator to configure reasons for requesting controlled prints and reprints of a document.
URS 5.17.12	The system shall provide the ability for a user to mark controlled copies as recalled.
URS 5.17.13	The system shall provide the ability for a user to view the list of controlled hard copies printed and there recall status for the purpose of destroying and replacing them.
URS 5.17.14	The system must provide the ability to audit the controlled print events (i.e., controlled printing, reprint, recall).
URS 5.17.15	The system shall provide the ability for an administrator to configure the ability to view and print the primary content (non-PDF version) of a document based on user roles, lifecycle status and document type.

Requirement ID	Description
URS 5.17.16	The system must provide the ability to print an expiry date on a printed copy.
URS 5.17.17	The system must provide the ability to watermark an indication such as "working copy" on a document.
URS 5.17.18	The system must provide the ability to generate sequential numbers to apply to printed master documents in order to generate issued prints.
URS 5.17.19	The system must provide the ability to apply overlays for any configured viewing or printing activity.
URS 5.17.20	The system must provide the ability to allow watermarking of different paper sizes including US letter, A4 and Legal.
URS 5.17.21	The system must provide the ability to allow watermarking of landscape and portrait paper orientations within the same document.
URS 5.17.22	The system shall provide the ability for an administrator to configure watermarks which may include any combination of document properties, fixed text strings, bar codes, images, the date and time of printing, and the user printing the document.
URS 5.17.23	The system must provide the ability to include watermarks in headers, footers and margins of the document.